

# Manual do Okteta

Friedrich W. H. Kossebau  
Alex Richardson  
Tradução: Marcus Gama



## Manual do Okteta

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>5</b>
<b>2</b>	<b>Fundamentos</b>	<b>6</b>
2.1	Iniciando o Okteta . . . . .	6
2.2	Uso . . . . .	6
<b>3</b>	<b>Ferramentas</b>	<b>7</b>
3.1	Resumo . . . . .	7
3.1.1	Analisadores e Manipuladores . . . . .	7
3.1.2	Ferramentas gerais . . . . .	8
3.2	Ferramenta de Estruturas . . . . .	8
3.2.1	Geral . . . . .	8
3.2.2	Instalar as definições das estruturas . . . . .	9
3.2.2.1	Instalar através do KNewStuff . . . . .	9
3.2.2.2	Instalar as definições das estruturas manualmente . . . . .	9
3.2.2.3	Usar as estruturas recém instaladas . . . . .	9
3.2.3	Compartilhar definições de estruturas . . . . .	9
3.2.4	Criar definições de estruturas . . . . .	10
3.2.5	Formato do arquivo XML da definição da estrutura . . . . .	10
3.2.6	Um exemplo de definição de estrutura em XML e JavaScript . . . . .	12
3.2.6.1	O passos comum, compartilhado por ambas as abordagens . . . . .	12
3.2.6.2	Uma definição de estrutura XML simples . . . . .	13
3.2.6.3	Uma estrutura simples em JavaScript . . . . .	13
3.2.6.4	Estruturas mais complexas . . . . .	14
3.2.6.5	Mais informações . . . . .	15
<b>4</b>	<b>Resumo da interface</b>	<b>16</b>
4.1	Itens dos menus . . . . .	16
4.1.1	Menu Arquivo . . . . .	16
4.1.2	O Menu Editar . . . . .	17
4.1.3	Menu Exibir . . . . .	17
4.1.4	O Menu Janelas . . . . .	19
4.1.5	O menu Favoritos . . . . .	19
4.1.6	Menu Ferramentas . . . . .	19
4.1.7	Menu Configurações . . . . .	19
<b>5</b>	<b>Créditos e licença</b>	<b>20</b>

### **Resumo**

O Okteta é um editor simples para os dados brutos dos arquivos. Este tipo de programa também é chamado de editor hexadecimal ou editor binário.

# Capítulo 1

## Introdução

O Okteta é um editor simples para os dados brutos dos arquivos.

Os dados são apresentados em duas variantes: como valores numéricos dos 'bytes' ou como os caracteres associados aos valores. Os valores e os caracteres poderão ser apresentados tanto em duas colunas (o modo tradicional de exibição em editores hexadecimais) quanto em colunas tendo o valor acima do caractere. A edição pode ser feita para ambos, os valores e os caracteres.

Além das funcionalidades normais de edição, o Okteta também oferece um pequeno conjunto de ferramentas, como uma decodificação da listagem em tabela para tipos de dados comuns, uma tabela que apresenta todos os 'bytes' possíveis juntamente com os seus equivalentes em valor e em caracteres, uma área de informação com estatísticas, uma calculadora de soma de verificação (checksum), uma ferramenta de filtragem e uma ferramenta de extração de texto.

Todas as modificações feitas aos dados carregados poderão ser desfeitas ou refeitas sem limites.

## Capítulo 2

# Fundamentos

### 2.1 Iniciando o Okteta

Digite **okteta** numa linha de comando ou selecione o **Editor Hexadecimal** do grupo **Aplicativos** → **Utilitários** no lançador de aplicativos.

Estão disponíveis as opções normais do Qt™ e do KDE Frameworks 5, que podem ser listadas digitando **okteta --help**.

As opções de linha de comando para o Okteta são:

<URL(s)> - abre os arquivos com as URLs indicadas

### 2.2 Uso

A janela principal do Okteta possui os seguintes componentes: a barra do menu, a barra de ferramentas, a barra de estado, as áreas de dados separadas por páginas e uma ou mais barras laterais com as ferramentas.

Quando é aberto um arquivo ou é criada uma sequência de 'bytes', os 'bytes' nela contidos são apresentados consecutivamente em linhas, tendo cada uma um determinado número de 'bytes'. Elas são apresentadas em duas variantes: como valores numéricos dos 'bytes' ou como os caracteres associados aos valores. Os valores e os caracteres poderão tanto ser apresentados separados em duas colunas como um após o outro, tendo o valor acima do caractere. Do lado esquerdo, aparece a posição do primeiro 'byte' de cada linha.

O tratamento é semelhante ao dos editores de texto: os dados poderão ser editados, cortados, copiados, colados, arrastados e soltos, como acontece num editor de texto. Existe um cursor que marca a posição atual. Ao pressionar a tecla **Insert**, poderá alternar entre os modos de sobreposição e inserção. O modo de sobreposição é mais restrito que nos editores de texto, uma vez que não permite nenhuma operação que altere o tamanho da sequência de 'bytes'.

Ao contrário dos editores de texto, o conteúdo aparece em duas variantes. Só uma destas está ativa no que diz respeito à inserção de dados. Existem dois cursores associados, presentes para o valor e para o caractere, sendo o cursor ativo o que está intermitente. Com os caracteres ativos, caracteres poderão ser inseridos como se faz num editor de texto normal. Com os valores ativos, ao digitar um número irá aparecer um editor mínimo para inserir o resto do valor.

A janela de pesquisa permite ao usuário procurar uma sequência específica de 'bytes', definida como valores (hexadecimal, decimal, octal, binário) ou texto (no momento, na codificação de 8 'bits', ou UTF-8).

Múltiplas sequências de 'bytes' podem ser abertas ao mesmo tempo, mas somente uma poderá estar ativa. Use o menu **Janelas** para selecionar a sequência de 'bytes' que está ativa.

## Capítulo 3

# Ferramentas

### 3.1 Resumo

O Okteta traz algumas ferramentas, algumas delas para analisar e manipular as sequências de 'bytes' e outras com fins mais gerais. Estas ferramentas poderão ser ativadas ou desativadas com o item **Ferramentas** do menu. Cada ferramenta tem uma área pequena que se associa a uma das barras laterais ou que flui livremente como uma janela. Você poderá acoplar, desacoplar, reorganizar e também empilhar as áreas de ferramentas com o mouse, clicando com o botão esquerdo do mouse na barra de título de uma área de ferramentas, movendo-o como desejar e soltando o botão esquerdo do mouse para completar a ação, ou, caso contrário, cancelando o movimento com a tecla **Esc**.

#### 3.1.1 Analisadores e Manipuladores

##### Tabela de valores/caracteres

A tabela apresenta todos os valores possíveis de um byte, quer na forma de caracteres ou em diferentes representações numéricas.

O valor selecionado poderá ser inserido na posição do cursor, repetida numa determinada quantidade de vezes. Isto poderá ser obtido usando-se o botão **Inserir** ou fazendo um duplo-clique na linha da tabela.

##### Filtro Binário

O filtro efetua operações binárias sobre os 'bytes' selecionados. Depois de selecionar a operação (E, OU, RODAR...), você poderá definir os parâmetros, se necessário, na área abaixo. O filtro é executado quando usar o botão **Filtro**.

##### Cadeias de caracteres

Esta ferramenta localiza os pedaços de texto nos 'bytes' selecionados. Depois de escolher o tamanho mínimo do texto, estes pedaços são pesquisados, bastando para isso usar o botão **Extrair**. A lista de textos apresentados poderá ser refinada se inserir um termo de filtragem.

##### Estatísticas

Esta ferramenta gera algumas estatísticas sobre a sequência de 'bytes'. As estatísticas fornecem a frequência de ocorrência de cada 'byte' na seleção. Ela pode ser calculada usando o botão **Construir**.

##### Soma de verificação

Esta ferramenta calcula diversos códigos de verificação (checksums ou hashsums) para os 'bytes' selecionados. Depois de escolher a operação e definir os parâmetros, se necessários, o somatório é calculado ao usar o botão **Calcular**.

### Tabela de Decodificação

A tabela apresenta o valor do 'byte' ou 'bytes' que começam no cursor, interpretado como um tipo de dados simples, como o Inteiro ou Real, mas também como UTF-8. Ao fazer duplo-clique sobre uma linha da tabela, será aberto um editor, de modo que o valor possa ser editado e alterado.

### Estruturas

Esta ferramenta ativa a investigação e edição das sequências de 'bytes', baseando-se nas definições de estruturas criadas pelo usuário. As instruções detalhadas sobre isto estão na sua própria [seção](#).

## 3.1.2 Ferramentas gerais

### Sistema de arquivos

Esta ferramenta oferece um navegador de arquivos incorporado que poderá usar para selecionar os arquivos a abrir.

### Documentos

Esta ferramenta mostra todos os arquivos carregados ou criados atualmente. Os símbolos marcam o arquivo com a área ativa e também mostram os arquivos que possuem alterações por salvar ou qual a cópia armazenada que foi modificada por outro programa.

### Favoritos

Esta ferramenta poderá ser usada para gerenciar os favoritos, como alternativa ao [menu Favoritos](#).

#### NOTA

Os favoritos são atualmente transitórios e não são salvos ao fechar uma lista de 'bytes' ou o programa em si.

### Informações do Arquivo

Esta ferramenta mostra algumas informações sobre o arquivo atual, incluindo o seu tipo, a localização do armazenamento e o tamanho.

### Terminal

Um Terminal integrado, onde a pasta de trabalho não está associada ao arquivo ativo.

### Conversão de codificações

A ferramenta reescreve os bytes de forma que os caracteres respectivos sejam os mesmos na outra codificação. Só há suporte para as codificações de 8 bits, sendo que os caracteres sem correspondência são, por enquanto, substituídos pelo valor fixo 0.

## 3.2 Ferramenta de Estruturas

### 3.2.1 Geral

A ferramenta de Estruturas permite a análise e edição de sequências de 'bytes' com base na definição de estruturas criadas pelo usuário, as quais poderão ser compostas por listas, uniões, tipos primitivos e enumerados.

Ela tem uma janela de configuração própria, que poderá ser acessada com o botão **Configurações**. Existem várias opções que poderão ser configuradas, como o estilo (decimal, hexadecimal ou



binário) no qual são apresentados os valores. Além disso, é possível escolher as definições de estruturas que são carregadas e quais são apresentadas na janela.

As estruturas são definidas nos arquivos de Definição de Estruturas do Okteta (baseados em XML, com a extensão `.osd`). Além disso, um arquivo `.desktop` contém os metadados sobre esse arquivo de descrição de estruturas, como o autor, o site e a licença.

Atualmente, não existe nenhum suporte incorporado para criar ou editar as definições das estruturas, assim elas deverão ser criadas manualmente, como se encontra descrito nas seções a seguir.

## 3.2.2 Instalar as definições das estruturas

### 3.2.2.1 Instalar através do KNewStuff

A forma mais simples de instalar novas definições de estruturas é usar o suporte incorporado do KNewStuff no Okteta. Para instalar uma estrutura existente, abra a janela de configuração da ferramenta de Estruturas. Aí, selecione a página de **Gerenciamento de estruturas** e pressione no botão **Obter estruturas novas....** A janela que aparecerá permite-lhe então instalar ou desinstalar estruturas.

### 3.2.2.2 Instalar as definições das estruturas manualmente

A ferramenta de Estruturas procura pelas descrições de estruturas na subpasta `okteta/structures/` da pasta pessoal do usuário para os dados dos aplicativos (descubra-a com o comando `qtpaths --paths LocalGenericoDeDados`). Talvez você precise criar essa pasta, caso ainda não tenha nenhuma definição de estrutura instalada.

Existem dois arquivos para cada definição de estrutura. Um arquivo para a definição em si e um arquivo `.desktop` para os metadados (autor, versão, etc).

Nessa pasta, existe uma subpasta para cada definição de estruturas, a qual contém tanto o arquivo `.desktop` como o `.osd` ou o `main.js` para essa definição.

Por exemplo, com a pasta de dados dos aplicativos `qtpaths --paths LocalGenericoDeDados` e com uma definição de estrutura chamada `EstruturaExemplo`, existe a pasta `okteta/structures/EstruturaExemplo`, que contém um arquivo `EstruturaExemplo.desktop` e um arquivo `EstruturaExemplo.osd`.

### 3.2.2.3 Usar as estruturas recém instaladas

Após ter instalado uma nova definição de estruturas, você poderá ter que reiniciar o Okteta antes de poder usá-lo. Após ter reiniciado o Okteta, abra a janela de configurações da ferramenta de Estruturas. Nela, selecione a página de **Gerenciamento das estruturas** e certifique-se que a definição da estrutura relevante é assinalada. Depois, mude para a página **Estruturas** e certifique-se de que o elemento desejado está listado no lado direito.

## 3.2.3 Compartilhar definições de estruturas

Para as estruturas comuns, você poderá não ter que criar você mesmo uma definição, mas sim reutilizar uma definição já existente de locais como o [store.kde.org](http://store.kde.org).

Você poderá também querer compartilhar você mesmo uma definição. Para fazer isso, crie um pacote de arquivos (por exemplo, um pacote compactado do tar, `.tar.gz`) contendo apenas a subpasta com o arquivo `.desktop` e o arquivo de definição da estrutura. No exemplo da última seção, seria a subpasta `EstruturaExemplo` com todo o seu conteúdo. Usando este formato para compartilhar definições de estruturas será possível instalá-las no Okteta e não irá exigir uma instalação manual.

### 3.2.4 Criar definições de estruturas

#### NOTA

Existe um guia mais atualizado, mas não completo, sobre a criação de definições de estruturas na [Wiki do KDE UserBase](#).

Existem duas maneiras diferentes de criar definições de estrutura. A primeira é escrever a definição em XML e a outra é usar JavaScript. A abordagem com JavaScript permite-lhe criar estruturas mais complexas com funcionalidades como, por exemplo, validação da estrutura. Usar XML oferece menos recursos, mas se uma estrutura estática é tudo que você precisa, pode ser a abordagem mais fácil. Se você precisar de uma estrutura dinâmica, ou seja, onde as dimensões de uma matriz dependem de outros valores na estrutura ou o layout da estrutura é diferente quando o valor de algum membro se altera, então você terá que escrever a definição da estrutura em JavaScript. Não há exceções a esta regra. Se você tiver uma matriz cujo comprimento deve ser **exatamente** o mesmo de acordo com outro valor na estrutura, então você pode usar XML. Mas se for algo como *comprimento - 1* deve-se usar JavaScript.

### 3.2.5 Formato do arquivo XML da definição da estrutura

#### NOTA

Existe um guia mais atualizado, mas não completo, sobre a criação de definições de estruturas na [Wiki do KDE UserBase](#).

O arquivo XML `.osd` possui um elemento raiz: `<data>` sem atributos. Dentro deste elemento deve existir um dos seguintes elementos:

#### `<primitive>`

Para criar tipos de dados primitivos, como por exemplo, `int` e `float`. Este elemento não aceita subelementos e pode ter os seguintes atributos:

##### **type**

O tipo deste tipo primitivo. Deve ser um dos seguintes:

- `char` para caracteres ASCII de 8 bits
- `int8`, `int16`, `int32`, `int64` para inteiros com sinal com os tamanhos respectivos
- `uint8`, `uint16`, `uint32`, `uint64` para inteiros sem sinal com os tamanhos respectivos
- `bool8`, `bool16`, `bool32`, `bool64` para booleanos sem sinal (0 = falso, qualquer outro valor = verdadeiro) com o tamanho respectivo
- `float` para números de vírgula flutuante IEEE754 de 32 bits
- `double` para números de vírgula flutuante IEEE754 de 64 bits

#### `<bitfield>`

Para criar um campo de bits. Este elemento não aceita subelementos e pode ter os seguintes atributos:

##### **width**

O número de bits usado por este campo de bits. Deve ser entre 1 e 64.

##### **type**

O tipo deste campo de bits. Deve ser um dos seguintes:

- *unsigned* para um campo de bits onde o valor será interpretada como um valor sem sinal (intervalo de 0 à  $2^{\text{width}} - 1$ )
- *signed* para o campo de bits onde o valor será interpretado como um valor com sinal (intervalo de  $-2^{\text{width} - 1}$  à  $2^{\text{width} - 1} - 1$ )
- *bool* para um campo de bits onde o valor será interpretado como um valor booleano

**NOTA**

Lembre-se sempre de adicionar um separador após um *<bitfield>*, uma vez que caso contrário o próximo elemento (exceto para textos e matrizes, que adicionam separadores automaticamente) iniciará no meio de um byte. Obviamente o separador não é necessário se você deseja este comportamento.

**<enum>**

Para criar um tipo primitivo, mas onde os valores são exibidos como membros de uma enumeração se possível. Este elemento não aceita subelementos (no entanto, você precisará de uma marca *<enumDef>* no arquivo para referenciá-lo). Ele tem os seguintes atributos:

**enum**

A enumeração subjacente para este valor. Deve corresponder ao atributo *name* de uma das marcas *<enumDef>* neste arquivo.

**type**

O tipo desta enumeração. Veja o atributo de tipo de *<primitive>*. A única diferença é que o *Double* e o *Float* não fazem sentido.

**<flags>**

Isto é o mesmo que *<enum>* com a única diferença sendo que os valores são representados como um *bitwise-or* de todos os valores da enumeração.

**<struct>**

Para criar uma estrutura. Todos os outros elementos (incluindo o *<struct>*) podem ser filhos deste e então serão parte da estrutura resultante

**<union>**

Para criar uma união. Basicamente, o mesmo que *<struct>* exceto pelo fato que todos os elementos filho iniciarão a partir do mesmo deslocamento. Útil para interpretar a mesma sequência de bytes de várias maneiras.

**<array>**

Para criar uma matriz. Este elemento aceita exatamente um filho (o tipo subjacente da matriz), que pode ser qualquer dos elementos, até mesmo a *<array>* em si. Ele possui os seguintes atributos:

**length**

O número de elementos nesta matriz como um número decimal. Alternativamente, ele pode também ser um texto que corresponde ao atributo nome de um elemento *<primitive>*, *<enum>* ou *<flags>* previamente definido. Então o comprimento será o valor deste elemento. Atualmente está limitado à 10000, porque matrizes maiores usariam muita memória e tornariam esta ferramenta muito lenta.

**<string>**

Para criar um texto em várias codificações. Por padrão, você obtém um texto terminado em *NULL* no estilo C. No entanto, diferentes tipos de texto podem ser criados com os seguintes atributos:

### **terminatedBy**

Este atributo determina com que ponto de código unicode o texto é terminado. Deve ser um número hexadecimal (opcionalmente com um prefixo *0x*). Quando a codificação for ASCII, somente valores até o *0x7f* são significativos. Se isto, *maxCharCount* ou *maxByteCount* estão definidos, isto é considerado como 0 (texto no estilo C)

### **maxCharCount**

O número máximo de caracteres que este texto pode ter. Se *terminatedBy* estiver definido então o que for atingido primeiro termina o texto. Isto é mutuamente exclusivo com o *maxByteCount*

### **maxByteCount**

O número máximo de bytes que o texto pode ter. Se *terminatedBy* estiver definido também, então o que for atingido primeiro termina o texto. Isto é mutuamente exclusivo com *maxCharCount*. Com codificações como *ASCII* isto é o mesmo que *maxCharCount*.

### **type**

A codificação do texto. Pode ser uma das seguintes:

- *ASCII*
- *LATIN-1*
- *UTF-8*
- *UTF-16-LE* ou *UTF-16-BE*. Se os sufixos *-LE* ou *-BE* não forem fornecidos, o little endian é considerado.
- *UTF-32-LE* ou *UTF-32-BE*. Se os sufixos *-LE* ou *-BE* não forem fornecidos, o little endian é considerado.

Cada elemento também aceita um atributo *name* que é então visível na visão de estruturas.

## **3.2.6 Um exemplo de definição de estrutura em XML e JavaScript**

### **NOTA**

Existe um guia mais atualizado, mas não completo, sobre a criação de definições de estruturas na [Wiki do KDE UserBase](#).

### **3.2.6.1 O passos comum, compartilhado por ambas as abordagens**

Nosso arquivo de metadados se parece com isto:

```
[Desktop Entry]
Icon=arrow-up<:\coref{1}{icon}>
Type=Service
ServiceTypes=KPluginInfo

Name=Estrutura simples de teste
Comment=Uma estrutura de teste simples contendo apenas dois itens

X-KDE-PluginInfo-Author=Alex Richardson
X-KDE-PluginInfo-Email=foo.bar@email.org
X-KDE-PluginInfo-Name=estruturasimples
X-KDE-PluginInfo-Version=1.0
X-KDE-PluginInfo-Website=https://www.plugin.org/
X-KDE-PluginInfo-Category=structure
X-KDE-PluginInfo-License=LGPL
X-KDE-PluginInfo-EnabledByDefault=false
```

- ❶ O ícone exibido no Okteta para esta estrutura. Pode ser qualquer coisa encontrada na execução do `kdiallog --geticon` ou o caminho para um ícone

Estes campos devem parecer bem auto-explicativos, exceto pelo `X-KDE-PluginInfo-Name`. O valor deste campo deve corresponder ao nome da pasta que contém o arquivo bem como o nome do arquivo `.desktop`. Ao criar as definições de estrutura em XML o nome do arquivo `.osd` deve também corresponder ao nome.

Neste exemplo, teríamos uma pasta chamada `estruturasimples` que contém os arquivos `estruturasimples.desktop`. Ao definir as estruturas em XML, a pasta iria conter também um arquivo chamado `estruturasimples.osd`. Se usarmos o JavaScript, iríamos ter um arquivo chamado `main.js` em alternativa.

### 3.2.6.2 Uma definição de estrutura XML simples

Para começar, vamos criar uma definição para uma estrutura de testes muito simples, contendo apenas tipos de dados primitivos (um caractere, um inteiro de 32 bits com sinal e um campo de bits). Isto poderia ser expresso em C/C++ como:

```
struct simple {
    char aChar;
    int anInt;
    bool bitFlag :1;
    unsigned padding :7;
};
```

O primeiro passo é escrever o arquivo `.osd` de acordo com o formato de arquivo definido na seção anterior. Nós o chamaremos de `estruturasimples.osd`:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <struct name="simple">
    <primitive name="aChar" type="Char"/>
    <primitive name="anInt" type="Int32"/>
    <bitfield name="bitFlag" type="bool" width="1"/>
    <bitfield name="padding" type="unsigned" width="7"/>
  </struct>
</data>
```

que é bastante similar a uma definição em C/C++.

Agora crie uma pasta `estruturasimples` dentro da pasta de instalação da estrutura (veja como instalar manualmente definições de estruturas) e copie os dois arquivos para esta pasta. Agora você pode reiniciar o Okteta, para poder usar a nova estrutura.

### 3.2.6.3 Uma estrutura simples em JavaScript

Para implementar a estrutura acima em JavaScript, crie um arquivo chamado `main.js` em vez de `estruturasimples.osd` e altere o `'X-KDE-PluginInfo-Category=structure'` para `'X-KDE-PluginInfo-Category=structure/js'`. O conteúdo desse arquivo deverá ser:

```
function init() {
    var estrutura = struct({
        caractere : char(),
        inteiro : int32(),
        opcaoBit : bitfield("bool", 1),
        preenchimento : bitfield("unsigned", 7),
```

```

    })
    return estrutura;
}

```

A estrutura apresentada pelo Okteta é sempre o valor devolvido pela função `init`.

As seguintes funções poderão ser invocadas para criar um tipo primitivo:

- `char()`
- `int8()`, `int16()`, `int32()` ou `int64()`
- `uint8()`, `uint16()`, `uint32()` ou `uint64()`
- `bool8()`, `bool16()`, `bool32()` ou `bool64()`
- `float()`
- `double()`

A função `'bitfield'` recebe dois parâmetros, sendo o primeiro um texto que consiste em `bool`, `signed` ou `unsigned`. O segundo parâmetro é um inteiro que define o tamanho em `'bits'`.

#### 3.2.6.4 Estruturas mais complexas

A seguir, iremos criar uma estrutura mais complexa, que irá se chamar "complexa" e salvá-la num arquivo `complexa.osd`. Esta estrutura irá conter duas listas (uma de tamanho fixo e outra com o tamanho determinado durante a execução), assim como uma estrutura encadeada e uma união.

```

<?xml version="1.0" encoding="UTF-8"?>
<data>
  <struct name="complexa">
    <primitive name="tamanho" type="UInt8" />
    <union name="umaUniao">
      <array name="quatroBytes" length="4">
        <primitive type="Int8" />
      </array>
    </union>
    <struct name="encadeado">
      <array name="texto" length="tamanho"> <!-- referencia o ←
        campo 'tamanho' acima -->
        <primitive type="Char" />
      </array>
    </struct>
  </struct>
</data>

```

Isto irá corresponder ao seguinte, em pseudo-C/C++

```

struct complexa {
    uint8_t tamanho;
    union umaUniao {
        int8_t quatroBytes[4];
    };
    struct encadeado {
        char texto[tamanho] //C++ não válido por referenciar o valor ←
        de 'tamanho'
    };
};

```

**NOTA**

Obviamente, você só poderá ter campos de referência de listas de tamanho dinâmico antes da lista.

A seguir, será criado o arquivo `complexo.desktop`, como foi feito no exemplo anterior (certifique-se de que define o `X-KDE-PluginInfo-Name` corretamente) e faça o mesmo para instalar ambos os arquivos.

### 3.2.6.5 Mais informações

Alguns exemplos de definições de estruturas poderão ser encontrados no [repositório Git](#). Isso inclui, por exemplo, o cabeçalho dos arquivos PNG e ELF. Um esquema em XML que descreve a estrutura do arquivo `.osd` pode ser encontrado [aqui](#). Se precisar de mais informações, sinta-se à vontade em contactar-me através do e-mail [arichardson.kde@gmail.com](mailto:arichardson.kde@gmail.com)

## Capítulo 4

# Resumo da interface

### 4.1 Itens dos menus

Além dos menus normais do KDE, descritos no capítulo [Menu](#) dos Fundamentos do KDE, o Okteta tem estes itens de menu específicos do aplicativo:

#### 4.1.1 Menu Arquivo

##### Arquivo → Novo (Ctrl+N)

Cria uma nova sequência de 'bytes'...

- **Vazio:** ... uma sequência vazia.
- **Da área de transferência:** ... com o conteúdo atual da área de transferência.
- **Padrão...:** ... com um determinado padrão.
- **Dados aleatórios...:** ... com dados aleatórios.
- **Sequência:** ... com todos os 'bytes' de 0 a 255.

##### Arquivo → Exportar

Exportar os bytes selecionados para um arquivo...

- **Valores:** ... codificados como valores de bytes. Por padrão, os valores são separados por um espaço em branco. Os caracteres de **Separação** podem ser alterados na caixa de diálogo **Exportar**.
- **Caracteres:** ... codificados como texto simples.
- **Base64:** ... codificado no formato [Base64](#).
- **Base32:** ... codificado no formato [Base32](#).
- **Ascii85:** ... codificado no formato [Ascii85](#).
- **Uuencoding:** ... codificado no formato [Uuencoding](#).
- **Xxencoding:** ... codificado no formato [Xxencoding](#).
- **Intel hexadecimal:** ... codificado no formato [Intel hexadecimal](#).
- **Registro-S:** ... codificado no formato [Registro-S](#).
- **Vetor em C:** ... definido como uma matriz na linguagem de programação C.
- **Ver em Texto Simples:** ... como na área de dados, com o deslocamento, os valores dos 'bytes' e os seus caracteres.



**Arquivo → Permissões → Definir apenas como leitura**

Quando estiver definido, as alterações não poderão ser feitas na sequência de dados carregada.

**Arquivo → Fechar todos os outros**

Fecha todas as sequências de 'bytes', exceto a atual.

## 4.1.2 O Menu Editar

**Editar → Copiar como**

Copia os bytes selecionados em um dos diferentes formatos para a área de transferência. Para obter uma lista dos formatos disponíveis, veja o item de menu **Arquivo → Exportar**

**Editar → Inserir**

**Inserir um padrão...**

Inserir uma sequência definida de 'bytes' na posição do cursor.

As opções na janela permitem-lhe definir o tamanho do padrão e o seu formato (Hexadecimal, Decimal, Octal, Binário ou Character(es) ou UTF-8).

**Editar → Desmarcar (Ctrl+Shift+A)**

Desmarca a seleção atual.

**Editar → Selecionar o intervalo... (Ctrl+E)**

Abre uma janela incorporada para definir o intervalo de seleção.

**Editar → Modo de Sobreposição (Ins)**

Comuta entre o modo de Inserção e o de Sobreposição.

**NOTA**

O modo de sobreposição está implementado de forma a ser bastante restrito; não é possível alterar o tamanho dos dados (não permite adicionar nem remover 'bytes').

**Editar → Localizar... (Ctrl+F)**

Procura um padrão definido no documento. Poderão ser procurados padrões em hexadecimal, decimal, octal, binário ou texto.

As opções da janela permitem-lhe definir o ponto inicial, a direção e a abrangência da pesquisa.

**Editar → Ir para a Posição... (Ctrl+G)**

Move o cursor para uma determinada posição.

## 4.1.3 Menu Exibir

**Ver → Mostrar a Posição da Linha (F11)**

Ativa ou desativa a apresentação da posição da linha numa área à esquerda.

**Ver → Mostrar os Valores ou Caracteres**

Selecione qual das interpretações de 'bytes' são apresentadas. Os valores possíveis são:

- **Valores**

- **Caracteres**
- **Valores e Caracteres**

**Ver → Codificação do Valor**

Seleciona a codificação dos valores como sendo:

- **Hexadecimal**
- **Decimal**
- **Octal**
- **Executável**

**Exibir → Codificação de caracteres**

Seleciona a codificação dos caracteres no submenu.

**Ver → Mostrar os Caracteres Invisíveis**

Ativa ou desativa a apresentação dos caracteres invisíveis. Se a visualização estiver desativada, no local correspondente da coluna de caracteres, é colocado um outro caractere de substituição.

**Exibir → Definir os bytes por linha**

Selecione os bytes mostrados por linha na caixa de diálogo; o valor padrão é 16 bytes.

**Exibir → Definir os bytes por grupo**

Por padrão, os valores hexadecimais são apresentados em grupos de 4 bytes. Usando este item do menu, você poderá adaptar isso de acordo com as suas preferências em uma caixa de diálogo.

**Ver → Layout dinâmico**

Define as regras de layout da área de dados. Isto define quantos 'bytes' são apresentados por linha, dependendo da largura da janela. As regras possíveis são:

- **Desligado:** O layout é fixo para o número atual de 'bytes' por linha e não se adapta à alteração do tamanho da janela.
- **Repartir apenas os grupos de 'bytes' completos:** Coloca o máximo de 'bytes' por linha possíveis, desde que fiquem grupos de 'bytes' completos.
- **Ligado:** O mesmo que o anterior, só que permitindo também grupos incompletos de 'bytes'.

**Ver → Modo de Visualização**

Seleciona o layout da janela como sendo:

- **Colunas:** A interpretação dos valores e caracteres é apresentada no layout clássico, com cada um deles apresentado numa coluna em separado.
- **Linhas:** A interpretação em caracteres dos 'bytes' é apresentada diretamente abaixo da interpretação do valor.

**Ver → Dividir na horizontal (Ctrl+Shift+T)**

Divide a área de visualização atualmente em primeiro plano em duas partes, adicionando uma cópia da área atual à outra área abaixo.

**Ver → Dividir na vertical (Ctrl+Shift+L)**

Divide a área de visualização atualmente em primeiro plano em duas partes, adicionando uma cópia da área atual à outra área à direita.

**Ver → Fechar a área de visualização (Ctrl+Shift+R)**

Fecha a área de visualização atualmente em primeiro plano.

#### **Ver → Ver perfil**

As configurações de visualização podem ser separadamente armazenadas em perfis de visão. O perfil atualmente selecionado pode ser atualizado diretamente a partir das configurações de visualização, ou um novo pode ser criado a partir destas configurações. Todos os perfis podem ser gerenciados em uma janela disponível em **Configurações → Gerenciar perfis de visão...**

### **4.1.4 O Menu Janelas**

Oferece uma lista com as janelas atuais. Selecione a janela ativa.

### **4.1.5 O menu Favoritos**

Vários favoritos podem ser definidos para uma única sequência de 'bytes'. Cada sequência tem o seu próprio conjunto de favoritos, sendo o conjunto apropriado apresentado no fundo do **Favoritos**. Escolha um favorito do menu para mover o cursor e a janela para o mesmo.

#### **NOTA**

Os favoritos são atualmente transitórios e não são salvos ao fechar uma lista de 'bytes' ou o programa em si.

#### **Favoritos → Adicionar aos Favoritos (Ctrl+B)**

Marca um determinado local na sequência de 'bytes'.

#### **Favoritos → Remover o Favorito (Ctrl+Shift+B)**

Remove o favorito atual. Este comando só está disponível se o cursor estiver num local marcado como favorito.

#### **Favoritos → Remover Todos os Favoritos**

Limpa a lista de favoritos.

#### **Favoritos → Ir para o Favorito Anterior (Alt+Cima)**

Move o cursor para o favorito anterior.

#### **Favoritos → Ir para o favorito seguinte (Alt+Baixo)**

Move o cursor para o favorito seguinte.

### **4.1.6 Menu Ferramentas**

Oferece uma lista com as ferramentas instaladas. Ativa ou desativa a apresentação de cada uma das ferramentas. Você poderá ver uma descrição mais detalhada de cada ferramenta na seção [Ferramentas](#).

### **4.1.7 Menu Configurações**

#### **Configurações → Gerenciar perfis de visão...**

Abre a janela para criar, editar, remover e configurar um perfil de visão padrão.

## Capítulo 5

# Créditos e licença

Okteta

Direitos autorais do programa 2006-2012 de Friedrich W. H. Kossebau [kossebau@kde.org](mailto:kossebau@kde.org)

Direitos autorais da documentação 2008, 2010 Friedrich W. H. Kossebau [kossebau@kde.org](mailto:kossebau@kde.org), Alex Richardson [arichardson.kde@gmail.com](mailto:arichardson.kde@gmail.com)

Tradução de Marcus Gama [marcus.gama@gmail.com](mailto:marcus.gama@gmail.com)

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

Este programa é licenciado sob os termos da [Licença Pública Geral GNU](#).