

El manual de l'Okteta

Friedrich W. H. Kossebau
Alex Richardson
Traductor: Josep M. Ferrer



El manual de l'Okteta

Índex

1	Introducció	5
2	Conceptes bàsics	6
2.1	Començant amb l'Okteta	6
2.2	Ús	6
3	Eines	7
3.1	Vista general	7
3.1.1	Analitzadors i manipuladors	7
3.1.2	Eines generals	8
3.2	Eina d'estructures	8
3.2.1	General	8
3.2.2	Instal·lació de les definicions d'estructura	9
3.2.2.1	Instal·lació utilitzant el KNewStuff	9
3.2.2.2	Instal·lació manual de les definicions d'estructura	9
3.2.2.3	Ús de les estructures instal·lades recentment	9
3.2.3	Compartint les definicions d'estructura	9
3.2.4	Creació de definicions d'estructura	9
3.2.5	Format del fitxer XML de definició d'estructura	10
3.2.6	Un exemple de definició d'estructura tant en XML com en JavaScript	12
3.2.6.1	El pas comú compartit per ambdós enfocaments	12
3.2.6.2	Una definició d'estructura XML senzilla	13
3.2.6.3	L'estructura senzilla en JavaScript	13
3.2.6.4	Estructures més complexes	14
3.2.6.5	Informació addicional	15
4	Vista general de la interfície	16
4.1	Elements del menú	16
4.1.1	El menú Fitxer	16
4.1.2	El menú Edita	17
4.1.3	El menú Visualitza	17
4.1.4	Menú de les finestres	18
4.1.5	Menú de les adreces d'interès	19
4.1.6	El menú Eines	19
4.1.7	El menú Arranjament	19
5	Crèdits i llicència	20

Resum

L'Okteta és un editor senzill per a les dades sense processar dels fitxers. Aquest tipus de programa també s'anomena editor hexadecimal o editor binari.

Capítol 1

Introducció

L'Okteta és un editor senzill per a les dades sense processar dels fitxers.

Les dades es mostren en dues variants: com a valors numèrics dels bytes i com a caràcters assignats als valors. Els valors i els caràcters es poden mostrar en dues columnes (la visualització tradicional en editors hexadecimal) o en files amb el valor a la part superior del caràcter. L'edició es pot fer tant pels valors com pels caràcters.

A més de les capacitats d'edició habituals, l'Okteta també porta un petit conjunt d'eines, com una taula que llista les descodificacions en tipus de dades simples habituals, una taula que llista tots els bytes possibles amb els seus equivalents en caràcter i valor, una vista d'informació amb una estadística, una calculadora de suma de verificació, una eina de filtre i una eina d'extracció de cadenes.

Totes les modificacions a les dades carregades es poden desfer o refer sense limitació.

Capítol 2

Conceptes bàsics

2.1 Començant amb l'Okteta

Teclegeu **okteta** en un indicador d'ordres o seleccioneu **Editor hexadecimal** del grup **Aplicacions** → **Utilitats** en el llançador d'aplicacions.

Són disponibles les opcions estàndard de la línia d'ordres de les Qt™ i dels Frameworks 5 del KDE, i es poden llistar introduint **okteta --help**.

Les opcions de la línia d'ordres específiques de l'Okteta són:

`<URL /s>` - obre fitxer/s des de l'URL/s especificat/s

2.2 Ús

La finestra principal de l'Okteta té els components següents: una barra de menús, una barra d'eines, una barra d'estat, una o diverses barres laterals amb eines, i la vista principal amb les vistes de dades amb pestanyes.

Quan s'obre un fitxer o es crea una matriu nova de bytes, els bytes continguts es llisten consecutivament en línies amb un nombre donat de bytes per línia. Es mostren en dues variants: com a valors numèrics dels bytes i com a caràcters assignats als valors. Els valors i els caràcters es poden mostrar separats en dues columnes o una al costat de l'altra amb el valor a la part superior del caràcter. Al costat esquerre es mostren els desplaçaments del primer byte de cada línia.

La gestió és similar a la de la majoria d'editors de text: les dades es poden editar, retallar, copiar, enganxar, arrossegat i deixar anar tant text com calgui. Un cursor marca la posició actual. Prement la tecla **Inser** es commuta entre els modes de sobreescritura i inserció. El mode de sobreescritura és més estricte que en els editors de text, ja que no permet cap operació que canviï la mida de la matriu de bytes.

Contràriament als editors de text, el contingut es mostra en dues variants. Només un d'ells està actiu respecte a l'entrada nova. Hi ha dos cursors enllaçats per la visualització dels valors i dels caràcters, el cursor actiu estarà parpellejant. Amb els caràcters actius, aquests es poden introduir com es fa als editors de text. Amb els valors actius, escriure un dígit obrirà un editor mínim per a introduir la resta del valor.

El diàleg de cerca permet a l'usuari cercar una cadena específica de bytes, definible com a valors (hexadecimal, decimal, octal, binari) o com a text (codificació actual de 8 bits o UTF-8).

Es poden obrir múltiples matrius de bytes al mateix temps, però només una pot estar activa. Utilitzeu el menú **Finestres** per a seleccionar quina matriu de bytes estarà activa.

Capítol 3

Eines

3.1 Vista general

L'Okteta porta algunes eines, algunes per a analitzar i manipular matrius de bytes i altres amb un propòsit més general. Aquestes eines es poden activar o desactivar des de l'entrada **Eines** a la barra de menús. Cada eina té una vista petita, que s'acobla en una de les barres laterals o flota lliurement com a finestra. Podeu acoblar, desacoblar, reorganitzar i també apilar les vistes d'eina amb el ratolí, prement el botó esquerre del ratolí sobre la barra de títol d'una vista d'eina, movent-la com vulgueu i alliberant el botó esquerre del ratolí per a completar l'acció, en cas contrari cancel·leu-la prement la tecla **Esc**.

3.1.1 Analitzadors i manipuladors

Taula de valors/caràcters

La taula llista tots els valors possibles dels bytes, com a caràcter i en les diferents codificacions numèriques.

El valor seleccionat es pot inserir a la posició del cursor per a un nombre definit de bytes. Això es pot aconseguir utilitzant el botó **Insereix** o fent doble clic a la línia de la taula.

Filtre binari

El filtre realitza operacions binàries en els bytes seleccionats. Després d'escollir l'operació (AND, OR, CIRCULA...) els paràmetres, si n'hi ha, es poden establir en el quadre de sota. El filtre s'executa en usar el botó **Filtre**.

Cadenes

Aquesta eina localitza les cadenes en els bytes seleccionats. Després d'escollir la longitud mínima de la cadena, les cadenes se cerquen en usar el botó **Extracte**. La llista de les cadenes mostrades es pot reduir introduint un terme de filtre.

Estadístiques

Aquesta eina construeix una estadística per als bytes seleccionats. L'estadística dona la freqüència de l'ocurrència de cada valor del byte en la selecció. Es pot calcular utilitzant el botó **Construeix**.

Suma de verificació

Aquesta eina calcula diverses sumes de verificació o sumes de resum per als bytes seleccionats. Després de triar l'algorisme i establir el paràmetre, si n'hi ha, la suma es calcula en usar el botó **Calcula**.

Taula de descodificació

La taula mostra els valors del byte o dels bytes que comencen al cursor per a alguns tipus de dades simples habituals com enter o coma flotant, però també UTF-8. Fer doble clic sobre una línia de la taula obre un editor, de manera que el valor es pot editar i canviar.

Estructures

Aquesta eina permet investigar i editar les matrius de bytes basades en les definicions d'estructura que pot crear l'usuari. Les instruccions detallades es troben en una [secció](#) pròpia.

3.1.2 Eines generals

Sistema de fitxers

Aquesta eina ofereix un navegador de fitxers incrustat que es pot utilitzar per a seleccionar fitxers a obrir.

Documents

Aquesta eina mostra tots els fitxers actualment creats o carregats. Els símbols marquen el fitxer amb la vista activa actual i també mostren quins fitxers tenen canvis sense desar o quina còpia d'emmagatzematge ha estat modificada per un altre programa.

Adreces d'interès

Aquesta eina es pot utilitzar per a gestionar les adreces d'interès, alternativament al menú [adreces d'interès](#).

NOTA

Les adreces d'interès només són transitòries i no es desen si tanqueu una matriu de bytes o tot el programa.

Informació del fitxer

Aquesta eina mostra informació sobre el fitxer actual, incloent-hi el seu tipus, la ubicació de l'emmagatzematge i la mida.

Terminal

Un terminal incrustat, el directori de treball no està associat amb el fitxer actiu.

Conversió del joc de caràcters

L'eina reescriu els bytes de manera que els caràcters respectius són els mateixos que amb l'altre joc de caràcters. Només s'admeten jocs de caràcters de 8 bits, i els caràcters no coincidents actualment se substitueixen per un valor codificat a 0.

3.2 Eina d'estructures

3.2.1 General

L'eina Estructures permet analitzar i editar matrius de bytes basades en definicions d'estructura que pot crear l'usuari, i que es poden construir a partir de matrius, unions, tipus primitius i valors enumerats.

Té un diàleg de configuració propi, al qual es pot arribar utilitzant el botó **Arranjament**. Hi ha diverses opcions que es poden configurar, com l'estil (decimal, hexadecimal o binari) en el qual es mostren els valors. A més, és possible triar quines definicions d'estructura es carreguen i quines estructures es mostren en la vista.

Les estructures es defineixen en fitxers de definició d'estructura de l'Okteta (basats en XML, amb l'extensió de fitxer `.osd`). A més, un fitxer `.desktop` conté metadades sobre aquest fitxer de descripció d'estructura, com ara autor, pàgina d'inici i llicència.

Actualment no hi ha suport integrat per a la creació o edició de definicions d'estructura, per tant, això ha de fer-se manualment com es descriu en les seccions següents.

3.2.2 Instal·lació de les definicions d'estructura

3.2.2.1 Instal·lació utilitzant el KNewStuff

La manera més fàcil d'instal·lar definicions noves d'estructura és utilitzant el suport integrat del KNewStuff a l'Okteta. Per a instal·lar una estructura existent obriu el diàleg de configuració de l'eina Estructura. Allà seleccioneu la pestanya **Gestió d'estructures** i premeu el botó **Obtén estructures noves....** El diàleg que es mostra ara us permet instal·lar i desinstal·lar estructures.

3.2.2.2 Instal·lació manual de les definicions d'estructura

L'eina Estructura cerca descripcions d'estructura en el subdirectori `okteta/structures/` del directori de l'usuari per a les dades del programa (que es troba executant `qtpaths --paths GenericDataLocation`). És possible que hàgiu de crear aquest directori si encara no hi ha cap definició d'estructura instal·lada.

Hi ha dos fitxers per a cada definició d'estructura: un fitxer per a la definició real i un fitxer `.desktop` per a les metadades (autor, versió, etc.).

En aquest directori hi ha un subdirectori per a cada definició d'estructura, que conté tant el fitxer `.desktop` com el fitxer `.osd` o `main.js` d'aquesta definició.

Per exemple, amb el directori de dades del programa `qtpaths --paths GenericDataLocation` i una definició d'estructura anomenada `ExampleStructure`, hi ha el directori `okteta/structures/ExampleStructure` que conté un fitxer `ExampleStructure.desktop` i un fitxer `ExampleStructure.osd`.

3.2.2.3 Ús de les estructures instal·lades recentment

Després d'instal·lar una definició nova d'estructura haureu de reiniciar l'Okteta abans de poder-la utilitzar. Un cop s'ha iniciat l'Okteta, obriu el diàleg de configuració de l'eina Estructura. Allà seleccioneu la pestanya **Gestió d'estructures** i assegureu-vos que la definició de l'estructura corresponent està marcada. Després canvieu a la pestanya **Estructures** i assegureu-vos que l'element desitjat està llistat a la dreta.

3.2.3 Compartint les definicions d'estructura

Per a les estructures normals és possible que no necessiteu crear una definició vosaltres mateixos, però en el seu lloc podeu reutilitzar una definició ja existent des de llocs com store.kde.org.

També és possible que vulgueu compartir una definició vos mateix. Per a fer-ho, creeu un arxiu de fitxer (p. ex. un arxiu tar comprimit amb zip, `.tar.gz`) que contingui només el subdirectori amb el fitxer `.desktop` i el fitxer de definició d'estructura. Mirant l'exemple de l'última secció aquest seria el directori `ExampleStructure` amb tot el seu contingut. Utilitzar aquest format per a compartir les definicions d'estructures permet instal·lar-les dins de l'Okteta i no requereix cap instal·lació manual.

3.2.4 Creació de definicions d'estructura

NOTA

Una guia més actualitzada, però no completada per a l'escriptura de les definicions d'estructura es pot trobar [en el Wiki UserBase de KDE](#).

Hi ha dues maneres diferents de crear definicions d'estructures. El primer és escriure la definició en XML, l'altre és utilitzant JavaScript. L'enfocament JavaScript permet crear estructures més complexes amb característiques com ara validar l'estructura. L'ús de l'XML dona menys característiques, però si tot el que necessiteu és una estructura estàtica, aquest pot ser l'enfocament més fàcil. Si es necessita una estructura dinàmica, és a dir, on les longituds de la matriu depenen d'altres valors de l'estructura o la disposició de l'estructura és diferent quan algun valor membre canvia, llavors haureu d'escriure la definició de l'estructura en JavaScript. Hi ha una excepció a aquesta regla: si teniu una matriu on se suposa que la longitud és **exactament** el mateix que un altre valor en l'estructura, llavors podeu utilitzar XML. Però si és una mica com *length - 1* ha de ser JavaScript.

3.2.5 Format del fitxer XML de definició d'estructura

NOTA

Una guia més actualitzada, però no completada per a l'escriptura de les definicions d'estructura es pot trobar [en el Wiki UserBase de KDE](#).

El fitxer XML `.osd` té un element arrel: `<data>` sense atributs. Dins d'aquest element ha d'haver-hi un dels elements següents:

`<primitive>`

Es pot crear un tipus de dades «primitive» com, p. ex., *int* o *float*. Aquest element no accepta subelements i pot tenir els atributs següents:

type

El tipus d'aquest tipus primitiu. Ha de ser un dels següents:

- *char* per a un caràcter ASCII de 8 bits
- *int8*, *int16*, *int32*, *int64* per a un enter amb signe d'aquesta mida
- *uint8*, *uint16*, *uint32*, *uint64* per a un enter sense signe d'aquesta mida
- *bool8*, *bool16*, *bool32*, *bool64* per a un booleà sense signe (0 = fals, qualsevol altre valor = cert) d'aquesta mida
- *float* per un nombre de coma flotant IEEE754 de 32 bits
- *double* per un nombre de coma flotant IEEE754 de 64 bits

`<bitfield>`

Per a crear un camp de bits. Aquest element no accepta subelements i pot tenir els atributs següents:

width

El nombre de bits que utilitza aquest camp de bits. Cal que sigui entre 1 i 64.

type

El tipus d'aquest camp de bits. Ha de ser un dels següents:

- *unsigned* per a un camp de bits on el valor s'interpretarà com un valor sense signe (interval de valors de 0 a $2^{\text{width}} - 1$)
- *signed* per a un camp de bits on el valor s'interpretarà com un valor amb signe (interval de valors des de $-2^{\text{width} - 1}$ a $2^{\text{width} - 1} - 1$)
- *bool* per a un camp de bits on el valor s'interpretarà com un valor booleà

NOTA

Recordeu sempre afegir un farciment després d'un *<bitfield>*, ja que en cas contrari l'element següent (excepte les cadenes i matrius, que afegeixen automàticament el farciment) començarà enmig d'un byte. Òbviament, no és necessari fer farciments si voleu aquest comportament.

<enum>

Per a crear un tipus primitiu, però on els valors es mostren com a membres d'una enumeració si és possible. Aquest element no accepta subelements (tot i que necessitareu una etiqueta *<enumDef>* al fitxer per a fer-hi referència). Té els atributs següents:

enum

L'enumeració subjacent per a aquest valor. Ha de coincidir amb l'atribut *name* d'una de les etiquetes *<enumDef>* en aquest fitxer.

type

El tipus d'aquesta enumeració. Vegeu l'atribut tipus de *<primitive>*. La diferència només és que *Double* i *Float* no tenen sentit.

<flags>

Això és el mateix que *<enum>* amb l'única diferència que els valors es representen com a *bitwise-or* de tots els valors de l'enumeració.

<struct>

Per a crear una estructura. Tots els altres elements (incloent-hi *<struct>*) poden ser un fill d'això i després formar part de l'estructura resultant

<union>

Per a crear una unió. Bàsicament el mateix que *<struct>* excepte el fet que tots els elements fills començaran des del mateix desplaçament. Útil per a interpretar la mateixa seqüència de bytes de diverses maneres.

<array>

Per a crear una matriu. Aquest element accepta exactament un fill (el tipus subjacent de la matriu), que pot ser qualsevol dels elements, fins i tot la mateixa *<array>*. També té els atributs següents:

length

El nombre d'elements d'aquesta matriu com a nombre decimal. També pot ser una cadena que coincideixi amb l'atribut «name» d'un element *<primitive>*, *<enum>* o *<flags>* definits prèviament. Llavors la longitud serà el valor d'aquest element. Actualment es limita a 10000, ja que les matrius més grans utilitzarien massa memòria i alentirien massa l'eina.

<string>

Per a crear una cadena en diverses codificacions. Per defecte obtindreu una cadena a l'estil C finalitzada amb *NULL*. No obstant això, es poden crear tipus de cadena diferents amb els atributs següents:

terminatedBy

Aquest atribut determina amb quin punt de codi Unicode finalitza la cadena. Ha de ser un nombre hexadecimal (opcionalment començant per *0x*). Quan la codificació és ASCII, només són significatius els valors fins a *0x7f*. Si no s'ha establert ni aquest ni *maxCharCount* ni *maxByteCount*, s'assumeix que s'estableix a 0 (cadena a l'estil C)

maxCharCount

El nombre màxim de caràcters que pot tenir aquesta cadena. Si també s'ha establert *terminatedBy*, llavors qualsevol cosa que arribi primer finalitza la cadena. Això és mútuament exclouent amb *maxByteCount*

maxByteCount

El nombre màxim de bytes que aquesta cadena pot tenir de longitud. Si també s'ha establert *terminatedBy*, llavors qualsevol cosa que arribi primer finalitza la cadena. Això és mútuament exclouent amb *maxCharCount*. Amb codificacions com *ASCII* això és el mateix que *maxCharCount*.

type

La codificació d'aquesta cadena. Pot ser una de les següents:

- *ASCII*
- *LATIN-1*
- *UTF-8*
- *UTF-16-LE* o *UTF-16-BE*. Si no es donen els sufixes *-LE* ni *-BE*, s'assumeix «little endian».
- *UTF-32-LE* o *UTF-32-BE*. Si no es donen els sufixes *-LE* ni *-BE*, s'assumeix «little endian».

Tots els elements també accepten un atribut *name* que és visible a la vista d'estructures.

3.2.6 Un exemple de definició d'estructura tant en XML com en JavaScript

NOTA

Una guia més actualitzada, però no completada per a l'escriptura de les definicions d'estructura es pot trobar [en el Wiki UserBase de KDE](#).

3.2.6.1 El pas comú compartit per ambdós enfocaments

El nostre fitxer de metadades hauria de ser així:

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

Name=Estructura simple de prova
Comment=Una estructura de prova senzilla que conté dos elements

X-KDE-PluginInfo-Author=Alex Richardson
X-KDE-PluginInfo-Email=foo.bar@email.org
X-KDE-PluginInfo-Name=simplestruct
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 La icona mostrada a l'Okteta per a aquesta estructura pot ser qualsevol cosa trobada mitjançant l'execució de **kdiallog --geticon** o un camí a una icona

Aquests camps haurien de ser força autoexplicatius, excepte *X-KDE-PluginInfo-Name*. El valor d'aquest camp ha de coincidir amb el nom del directori que conté el fitxer així com el nom del fitxer *.desktop*. En crear les definicions d'estructura XML el nom del fitxer *.osd* també ha de coincidir amb el nom.

En aquest exemple tindríem un directori anomenat *simplestruct* que conté el fitxer *simplestruct.desktop*. En definir estructures en XML el directori també conté un fitxer anomenat *simplestruct.osd*. Utilitzant JavaScript en el seu lloc tindríem un fitxer anomenat *main.js*.

3.2.6.2 Una definició d'estructura XML senzilla

Per a començar creem una definició per a una estructura de prova molt senzilla que conté només tipus de dades integrals (un caràcter, un enter amb signe de 32 bits i un camp de bits). Això s'expressa en C/C++ com:

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

El primer pas és escriure el fitxer `.osd` d'acord amb el format de fitxer definit a la secció anterior. Invocarem `simplestruct.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 és força similar a la definició en C/C++.

Ara creeu un directori `simplestruct` dins del directori d'instal·lació de l'estructura (vegeu manualment les definicions d'estructura) i copieu els dos fitxers a aquest directori. Ara podeu reiniciar l'Okteta i utilitzar l'estructura nova.

3.2.6.3 L'estructura senzilla en JavaScript

Per a implementar l'estructura anterior en JavaScript, creeu un fitxer anomenat `main.js` en lloc de `simplestruct.osd` i canvieu `X-KDE-PluginInfo-Category=structure` a `X-KDE-PluginInfo-Category=structure/js`. El contingut d'aquest fitxer hauria de ser:

```
function init() {
    var structure = struct({
        aChar : char(),
        anInt : int32(),
        bitFlag : bitfield("bool", 1),
        padding : bitfield("unsigned", 7),
    })
    return structure;
}
```

L'estructura mostrada per l'Okteta és sempre el valor de retorn de la funció `init`.

Les funcions següents es poden anomenar per a crear un tipus primitiu:

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

- float()
- double()

La funció `camp de bits` té dos paràmetres, el primer és una cadena que consta d'un `bool`, `signed` o `unsigned`. El segon paràmetre és un enter que estableix l'amplada en bits.

3.2.6.4 Estructures més complexes

A continuació creem una definició d'una estructura més complexa que anomenarem «complex» i desarem en un fitxer anomenat `complex.osd`. Aquesta estructura contindrà dues matrius (una amb longitud fixa i una altra on la longitud es determina en temps d'execució) així com una estructura imbricada i una unió.

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
  <struct name="complex">
    <primitive name="size" type="UInt8" />
    <union name="aUnion">
      <array name="fourBytes" length="4">
        <primitive type="Int8" />
      </array>
    </union>
    <struct name="nested">
      <array name="string" length="size"> <!-- fa referència a la mida del camp anterior -->
        <primitive type="Char" />
      </array>
    </struct>
  </struct>
</data>
```

Això correspondria al següent en pseudocodi C/C++

```
struct complex {
    uint8_t size;
    union aUnion {
        int8_t fourBytes[4];
    };
    struct nested {
        char string[size] //no és C++ vàlid, fa referència al valor de la mida d'«uint8»
    };
};
```

NOTA

Òbviament només podeu tenir camps de referència de matrius de longitud dinàmica abans de la matriu.

A continuació creem el fitxer `complex.desktop` com en l'exemple anterior (assegureu-vos que heu establert `X-KDE-PluginInfo-Name` correctament) i també feu el mateix per a instal·lar els dos fitxers.

3.2.6.5 Informació addicional

Es poden trobar algunes definicions d'estructura d'exemple al [repositori Git](#). Això inclou per exemple la capçalera de fitxer per a fitxers PNG i la capçalera de fitxer ELF. Es pot trobar un esquema XML que descriu l'estructura del fitxer `.osd` [aquí](#). Si es necessita més informació us podeu posar en contacte amb mi a arichardson.kde@gmail.com

Capítol 4

Vista general de la interfície

4.1 Elements del menú

A més dels menús comuns del KDE descrits al capítol [Menú](#) en els Fonaments del KDE, l'aplicació Okteta disposa d'aquestes entrades de menú específiques:

4.1.1 El menú Fitxer

Fitxer → Nou (Ctrl+N)

Crear una matriu nova de bytes...

- **Buida:** ...com una de buida.
- **Des del porta-retalls:** ...pel contingut actual del porta-retalls.
- **Patró...:** ...amb un patró donat.
- **Dades aleatòries...:** ...amb dades aleatòries.
- **Seqüència:** ...amb tots els bytes de 0 a 255.

Fitxer → Exporta

Exporta els bytes seleccionats a un fitxer...

- **Valors:** ...codificats com a valors de bytes. De manera predeterminada els valors estan separats per un espai en blanc. Els caràcters de **separació** es poden canviar en el diàleg d'**Exportació**.
- **Caràcters:** ...codificat com a text net.
- **Base64:** ...codificat en el format [Base64](#).
- **Base32:** ...codificat en el format [Base32](#).
- **Ascii85:** ...codificat en el format [Ascii85](#).
- **Uuencoding:** ...codificat en el format [Uuencoding](#).
- **Xxencoding:** ...codificat en el format [Xxencoding](#).
- **Intel Hex:** ...codificat en el format [Intel Hex](#).
- **S-Record:** ...codificat en el format [S-Record](#).
- **Matriu C:** ...definida com una matriu en el llenguatge de programació C.
- **Visualitza en text net:** ...com en la vista de dades amb desplaçament, valors de bytes i caràcters.

Fitxer → Permisos → Estableix només de lectura

Quan s'estableix, els canvis no es poden fer a la matriu de bytes carregada.

Fitxer → Tanca totes les altres

Tanca totes matrius de bytes excepte l'actual.

4.1.2 El menú Edita

Edita → Copia com a

Còpia els bytes seleccionats en un dels diferents formats al porta-retalls. Per a una llista dels formats disponibles vegeu l'element de menú **Fitxer → Exporta**

Edita → Insereix

Insereix un patró...

Insereix una cadena de bytes especificada al cursor.

Les opcions en el quadre de diàleg permeten especificar el nombre d'insercions del patró i el format d'aquest (Hexadecimal, Decimal, Octal, Binari, Caràcter/s o UTF-8).

Edita → Deselecciona (Ctrl+Maj+A)

Deselecciona la selecció actual.

Edita → Selecciona un interval... (Ctrl+E)

Obre un diàleg incrustat per a introduir l'interval a seleccionar.

Edita → Mode sobreescriure (Insert)

Commuta entre el mode d'inserció i el mode de sobreescriptura.

NOTA

El mode de sobreescriptura s'implementa per a ser molt estricte, no és possible canviar la mida de les dades (no afegeix ni elimina bytes).

Edita → Cerca... (Ctrl+F)

Troba un patró especificat en el document. Es poden cercar patrons hexadecimal, decimal, octal, binari o de text.

Les opcions del quadre de diàleg us permeten especificar el punt d'inici, la direcció i l'interval de la cerca.

Edita → Ves al desplaçament... (Ctrl+G)

Mou el cursor a un desplaçament especificat.

4.1.3 El menú Visualitza

Visualitza → Mostra el desplaçament de la línia (F11)

Commuta mostrar el desplaçament de la línia en una subfinestra a l'esquerra activada i desactivada.

Visualitza → Mostra els valors o els caràcters

Selecciona quina de les interpretacions dels bytes es mostren. Són possibles:

- **Valors**
- **Caràcters**
- **Valors i caràcters**

Visualitza → Codificació dels valors

Selecciona la codificació dels valors de:

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

- **Binari**

Visualitza → Codificació dels caràcters

Selecciona la codificació dels caràcters des del submenú.

Visualitza → Mostra els caràcters no imprimibles

Alterna la visualització de caràcters no imprimibles activats i desactivats. Si la visualització està desactivada, als llocs corresponents de la columna del caràcter es mostra un caràcter substitutiu.

Visualitza → Defineix els bytes per línia

Selecciona els bytes mostrats per línia del diàleg, el valor per defecte és de 16 bytes.

Visualitza → Defineix els bytes per grup

Per defecte els valors hexadecimal es mostren en grups de 4 bytes. Usant aquest element de menú el podeu adaptar a les vostres preferències en un diàleg.

Visualitza → Disposició dinàmica

Estableix les regles per a la disposició de la visualització de dades. Això defineix quants bytes es mostren per línia, depenent de l'amplada de la vista. Les regles possibles són:

- **Inactiva:** La disposició és fixa al nombre actual de bytes per línia i no s'adapta al canvi de la mida de la vista.
- **Ajusta només els grups de bytes complets:** Posa tants bytes per línia com sigui possible, sempre que els grups de bytes estiguin complets.
- **Activa:** Igual que l'anterior, però també permet grups incomplets de bytes.

Visualitza → Mode de vista

Selecciona la disposició de la vista de:

- **Columnes:** Els valors i les interpretacions dels caràcters es mostren en la disposició clàssica amb cada un llistat en una columna separada.
- **Files:** La interpretació dels caràcters d'un byte es mostra directament sota la interpretació del valor.

Visualitza → Divideix horitzontalment (Ctrl+Maj+T)

Divideix l'àrea de visualització amb la vista actualment enfocada en dues parts i afegeix una còpia de la vista actual a l'àrea inferior nova.

Visualitza → Divideix verticalment (Ctrl+Maj+L)

Divideix l'àrea de visualització amb la vista actualment enfocada en dues parts i afegeix una còpia de la vista actual a l'àrea dreta nova.

Visualitza → Tanca l'àrea de visualització (Ctrl+Maj+R)

Tanca l'àrea de visualització amb la vista actualment enfocada.

Visualitza → Perfil de vista

La configuració de la vista es pot desar per separat com a perfils de vista. El perfil actualment seleccionat es pot actualitzar directament des de la configuració actual de la vista, o es pot crear un de nou a partir d'ells. Tots els perfils de visualització es poden gestionar en un diàleg disponible des d'**Arranjament → Gestiona els perfils de visualització...**

4.1.4 Menú de les finestres

Proporciona una llista de les vistes actuals. Seleccioneu la finestra activa.

4.1.5 Menú de les adreces d'interès

Es poden establir múltiples adreces d'interès per a una matriu d'un sol byte. Cada matriu de bytes té el seu propi conjunt d'adreces d'interès, i el conjunt apropiat es mostra a la part inferior del menú **Adreces d'interès**. Trieu una adreça d'interès del menú per a moure-hi el cursor i la vista.

NOTA

Les adreces d'interès només són transitòries i no es desen si tanqueu una matriu de bytes o tot el programa.

Adreces d'interès → Afegeix una adreça d'interès (Ctrl+B)

Afegeix una ubicació en la matriu de bytes.

Adreces d'interès → Elimina l'adreça d'interès (Ctrl+Maj+B)

Elimina l'adreça d'interès actual. Aquesta ordre només està disponible si el cursor es troba en una ubicació apuntada.

Adreces d'interès → Elimina totes les adreces d'interès

Neteja la llista d'adreces d'interès.

Adreces d'interès → Ves a l'adreça d'interès anterior (Alt+Fletxa amunt)

Mou el cursor a l'adreça d'interès anterior.

Adreces d'interès → Ves a l'adreça d'interès següent (Alt+Fletxa avall)

Mou el cursor a l'adreça d'interès següent.

4.1.6 El menú Eines

Proporciona una llista de les eines instal·lades. Activa o desactiva la visualització de cada eina. Una descripció detallada de cada eina que trobeu a la secció [Eines](#).

4.1.7 El menú Arranjament

Arranjament → Gestiona els perfils de visualització...

Obre un diàleg per a crear, editar, suprimir i establir un perfil de vista predeterminat.

Capítol 5

Crèdits i llicència

Okteta

Copyright del programa 2006-2012 Friedrich W. H. Kossebau kossebau@kde.org

Copyright de la documentació 2008, 2010 Friedrich W. H. Kossebau kossebau@kde.org, Alex Richardson arichardson.kde@gmail.com

Traductor de la documentació: Josep M. Ferrer txemaq@gmail.com

Aquesta documentació està llicenciada d'acord amb les clàusules de la [Llicència de Documentació Lliure de GNU](#).

Aquest programa està llicenciat d'acord amb les clàusules de la [Llicència Pública General de GNU](#).