

ディスク割り当て評価ツール v1.0.0

使用説明書

日立製作所

目次

1. はじめに	- 3 -
1.1. 目的	- 3 -
1.2. プロジェクト概要とスコープ	- 3 -
2. DAV 概要	- 4 -
2.1. DAV とは	- 4 -
2.2. 前提条件	- 4 -
3. DAV 動作環境構築手順	- 4 -
3.1. ディレクトリ構成	- 4 -
3.2. インストール手順	- 5 -
3.2.1. 前準備	- 5 -
3.2.2. インストール	- 5 -
4. DAV 使用手順	- 6 -
4.1. 提供コマンド概要	- 6 -
4.2. dac 使用方法	- 8 -
4.3. dav 使用方法	- 13 -
4.4. fbksinfo モジュールについて	- 22 -

1. はじめに

1.1. 目的

サーバ分野において、Linux を適用したシステムが普及・拡大している。数年前までは、Web サーバやメールサーバ、ネームサーバといったネットワークのフロントエンドサーバへの適用が中心であったが、最近では、アプリケーションサーバ、DB サーバから構成されるエンタープライズシステムへの適用ニーズも出てきている。サーバ分野において、Linux を適用したシステムが普及・拡大している。数年前までは Web サーバやメールサーバ、ネームサーバといったネットワークのフロントエンドサーバへの適用が中心であったが、最近ではアプリケーションサーバ、DB サーバから構成されるエンタープライズシステムへの適用ニーズも出てきている。

エンタープライズシステムでは迅速な障害対応が求められるが、Linux にはダンプやトレースといった障害解析のための標準的なツールが無く、障害発生時は、各社固有のノウハウで対応しているのが現状である。障害の中でも特に Linux ファイルシステムのディスク割り当てに関しては、フラグメンテーションが起こりにくく障害の原因にはなりにくいといわれており、評価するためのツールもほとんど提供されていない。

しかし最近では、データのアクセス性能が落ちる障害が多々見受けられるようになってきている。この原因のひとつとしてディスクのフラグメンテーションが考えられるが、評価ツールがないために問題を特定することができない。また、ディスク全体の割り当て状況の評価するツールを開発した場合でも、特定ファイルのアクセス性能が落ちる障害の場合には問題を特定できない。

ディスク割り当て評価ツール（以降 DAV と称する）ではこれらの問題を解決するため、ディスク割り当て状況を容易に取得/可視化でき、更にファイルを対象としたディスク割り当て状況の可視化も可能とすることを目的として開発を行った。

1.2. プロジェクト概要とスコープ

DAV の適用範囲を表 1-1 に示す。

表 1-1 DAV 適用範囲

項番	項目	マクロ名	備考
1	ハードウェア条件	PC: IA-32 をベースとした PC CPU: Pentium メモリ: 64MB 以上	
2	Linux カーネル	Upstream Kernel v2.6.9 Miracle Linux 3.0 (Asianux 1.0): v2.4.21	
3	ライブラリ	GTK+1.2	
4	ファイルシステム	ext2、ext3	

2. DAV 概要

2.1. DAV とは

DAV は Linux の ext2 / ext3 ファイルシステムのフラグメンテーション状況を視覚化して表示するツールであり、下記の機能を持つ。

- ファイルシステムのマウント状態によらず、フラグメンテーション状況を取得できる。
- フラグメンテーション取得対象として、パーティション全体 / 任意のディレクトリ下の全ファイル / 単一ファイルのいずれかを指定できる。

DAV はフラグメンテーション情報を取得する `dac` (`Disk Allocation Checker`) と、その情報を GUI 表示するラッパーの `dav` (`Disk Allocation Viewer`) の 2 つのプログラムおよび、マウント状態のパーティションのフラグメンテーション情報を取得する `fbksinfo` モジュールによって構成される。フラグメンテーション状況を取得する `dac` は単体でも動作し、フラグメンテーション状況をテキスト形式で出力させることができる。

DAV はカーネルを Upstream カーネル 2.6.9 に変更した Fedora Core 2 および、Miracle Linux V3.0 上で動作を確認している。

2.2. 前提条件

DAV のコンパイルおよびインストールに必要な条件を下記に示す。

- カーネルバージョンが 2.4 以上であること。
- カーネルソースが展開されていること。
- GTK+1.2 がインストールされていること。

3. DAV 動作環境構築手順

3.1. ディレクトリ構成

アーカイブを下記の手順で展開すると、`dav` のディレクトリツリーが展開される。(以降、`dav` ディレクトリ)

```
$ tar jxvf dav-XXX.tar.bz2
```

XXX は、`dav` バージョン

`dav` ディレクトリ下の主な内容を下記に示す。

dav/		
+----	README	概要、インストール方法
+----	COPYING	GPL ライセンス
+----	Makefile	メイクファイル
+----	path_list	パス設定ファイル
+----	doc/	ドキュメント
+----	src/	ソースファイル
	+---- common/	共通ソースファイル
	+---- dac/	dac ソースファイル
	+---- dav/	dav ソースファイル
	+---- drv/	fbksinfo モジュールソースファイル

3.2. インストール手順

3.2.1. 前準備

DAV では、カーネルソースおよびインストール先のディレクトリは dav ディレクトリ直下の path_list 中で決まっており、変更する場合は直接 path_list を書き換える必要がある。各種ディレクトリの定義マクロ名と、デフォルトのディレクトリを表 3-1 に示す。

表 3-1 各指定マクロ一覧

項番	項目	マクロ名	デフォルトディレクトリ
1	カーネルソース	KERN_DIR	/usr/src/linux-XXXX
2	実行ファイルインストール先	BIN_DIR	/usr/local/bin
3	fbksinfo モジュールインストール先	DRV_DIR	/lib/modules/XXXX/kernel/drivers/fbksinfo
4	マニュアルインストール先	MAN_DIR	/usr/local/man

XXXX : uname -r の出力文字列

3.2.2. インストール

インストールは、下記の手順で行う。

- (1). dav ディレクトリに移動

```
$ cd dav
```

- (2). ビルド

```
$ make
```

- (3). インストール (スーパーユーザで実行)

```
# make install
```

4. DAV 使用手順

4.1. 提供コマンド概要

DAV が提供するコマンドの一覧を下記に示す。

- dac フラグメンテーション状況取得とその出力（テキスト形式）を行う
- dav 内部で dac を起動し、その出力を GUI 表示する

dac / dav を実行する上での留意点・制限事項を下記に示す。

- スーパーユーザ権限で実行する必要がある。
- コマンド実行中に、フラグメンテーション状況取得対象パーティションのマウント状態を変更しないこと。
- アンマウントパーティションのフラグメンテーション状況を取得する場合、クリーンアンマウント状態であること。

DAV では、ファイルブロックをファイル構成順に見て行き、下記の状態の場合に、フラグメントブロックと判断する。

- (1). ファイルブロックが、システムブロックを挟んでいる。
- (2). ファイルブロックが、物理的に連続していない。
- (3). ファイルブロックの順番が逆転している。

フラグメントブロックの例（ファイルブロック数=6）を、図 4-1 に示す。

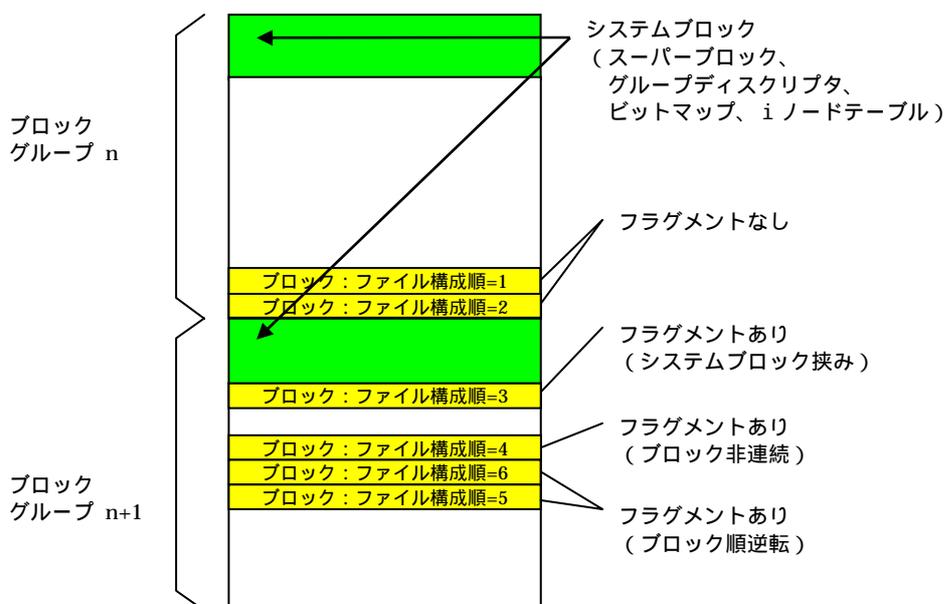


図 4-1 フラグメントブロック例

(1)のシステムブロックによるフラグメントは不可避であるため、他のフラグメントとは区別し、別に計数している。

DAVでのフラグメンテーション状況のテキスト形式出力/GUI表示では、以下の2つのブロック表示方法がある。

- (1). 物理的なブロック番号順でシステムブロックを含めて表示する表示。(ブロック番号順表示)
- (2). 確認対象のファイルを構成するブロックだけを、その構成順に表示。(ファイル構成順表示)

図 4-1 の例を2つの表示方法でそれぞれ表示した例を図 4-2 に示す。

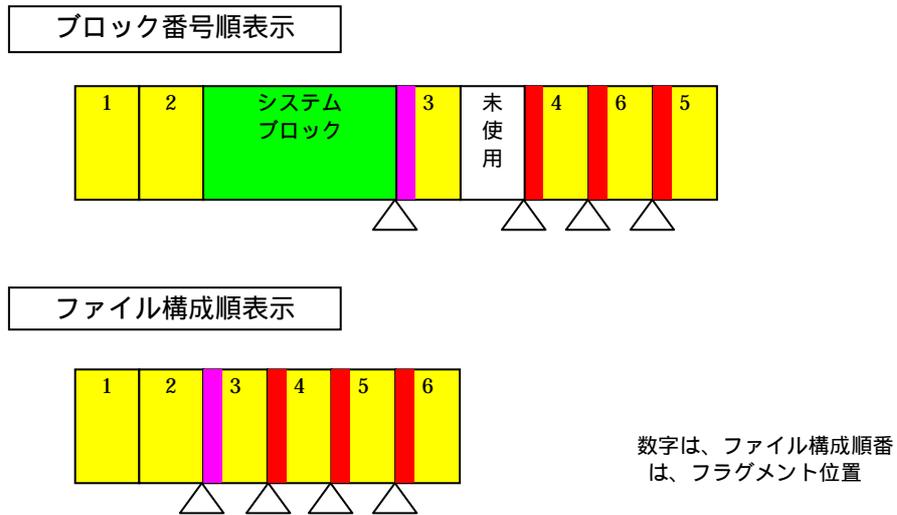


図 4-2 ブロック番号順/ファイル構成順での表示例

どの場合にどちらの方法で表示するかの概要は、表 4-1 を参照。また、どちらの表示の場合でも、フラグメントの判定方法は変わらない。

表 4-1 確認対象とブロック表示方法

項番	確認対象	付加情報	ブロック表示	備考
1	パーティション全体	-	ブロック番号順	
2	ディレクトリ	-	ブロック番号順	
3	ファイル	-	ブロック番号順 / ファイル構成順	選択可能
4	上記すべて	フラグメントファイル情報	ファイル構成順	

4.2. dac 使用方法

dac は、パーティション全体 / 任意のディレクトリ下の全ファイル / 単一ファイルのいずれかのフラグメンテーション状態を取得し、標準出力にテキスト形式で出力する。また、パーティションの一覧をマウント状態と合わせて表示することができる。

dac コマンドの説明を表 4-2 に示す。

表 4-2 dac コマンド説明

コマンド	dac	
書式	dac [-cChnstTv[v[v]]] [devname [path]]	
引き数	devname	デバイス名。デバイス名が指定されない場合は、システム内のパーティション一覧を /proc/partitions から読み出し、マウント状態と合わせて表示する。この時 /proc/partitions に含まれないデバイスに関しては表示しない
	path	パーティション devname 内のディレクトリまたはファイル名。先頭に / (スラッシュ) が必要。path が指定されない場合は、パーティション全体のフラグメント状況を表示する
オプション	c	出力をエスケープシーケンスを用いたカラー表示にする
	C	ファイルシステム中のブロック / inode ビットマップと dac 解析結果を比較チェックする。(デバッグ用)
	h	コマンドのヘルプを表示する
	n	ディレクトリをチェックする場合、そのディレクトリだけのフラグメント状況を表示し、下位のディレクトリおよびファイルのフラグメント状況は含めない。このオプションを指定しない場合は、下位のディレクトリおよびファイルのフラグメント状況を含めて表示する
	s	マウントパーティション中のフラグメント状況を確認する際に、fbksinfo モジュールを使用しない。fbksinfo については「4.4 fbksinfo モジュールについて」参照
	t	ブロック情報出力フォーマットを、ブロック種別 + 同一種別連続数のカンマ区切り文字列にする。カラム数 80 で行を折り返す。ブロック種別については後述の dac 出力例を参照
	T	ブロック情報出力を折り返さずに 1 行で全て表示する以外は、t オプションと同じ
	v	詳細情報表示レベル。v、vv、vvv と 3 つまで指定でき、v の数が多いほど詳細な情報を表示する。vv、vvv はデバッグ用

dac をオプション -Tv[Cns] で実行した出力をファイルに保存しておけば、いつでもそれを dav に読み込ませて GUI 表示することが可能である。

dac 実行手順を下記に示す。

- (1). スーパーユーザでログイン
- (2). fbksinfo モジュールをロード (一度ロードすれば、次回の dac / dav 実行時は不要)

```
# /sbin/modprobe fbksinfo
```

- (3). dac 実行

```
# dac (必要に応じてオプションおよび引数を指定)
```

dac の出力例とその見方を図 4-3 ~ 図 4-6 に示す。

----- partitions information -----		
name	mount	#blocks
/dev/hda	unmount	58593840
/dev/hda1	mount	12289693
/dev/hda2	unmount	1020127
/dev/hda3	unmount	1
/dev/hda5	unmount	11727418
/dev/hda6	unmount	3911796
/dev/hda7	unmount	2353491
/dev/hda8	unmount	240943
/dev/hda9	unmount	56196
/dev/hda10	unmount	2353491
/dev/hda11	unmount	240943
/dev/hda12	unmount	56196
/dev/hda13	unmount	8001

execute "dac -h" for help

} パーティション一
覧ヘッダ

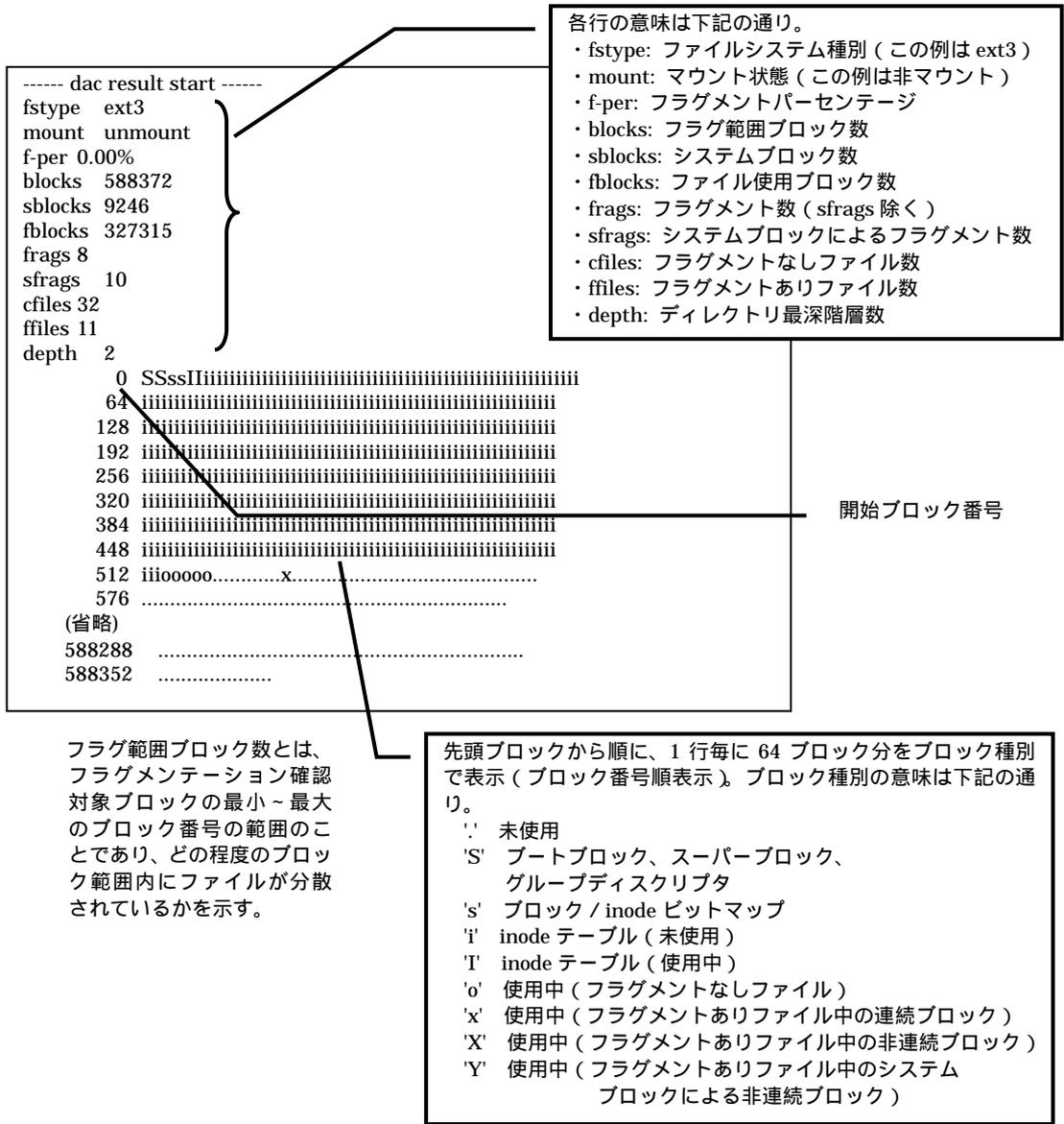
デバイス名

マウント状態

ブロック数

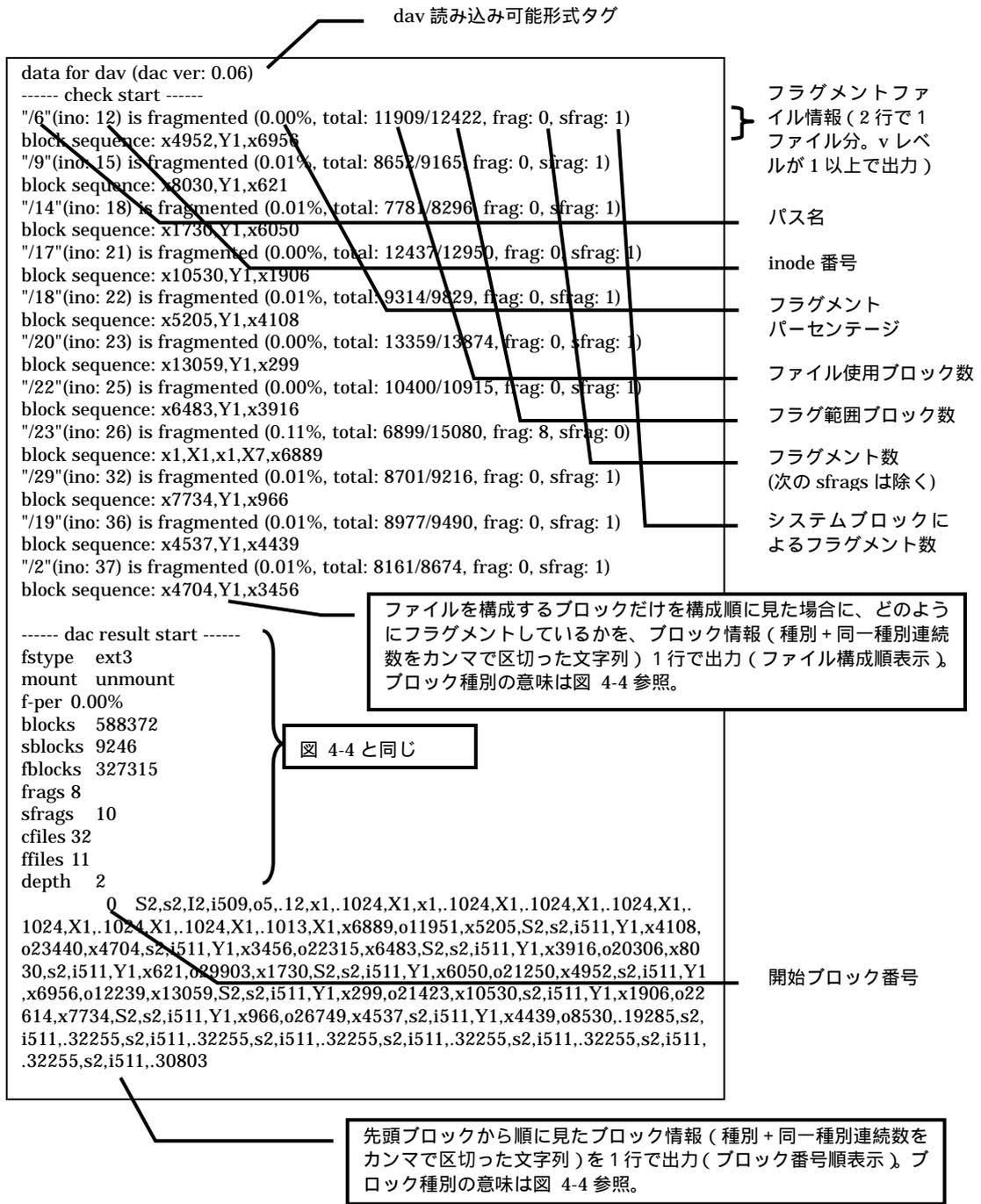
dac(オプションなし) の出力例

図 4-3 dac 出力例 (1 / 4) - パーティション一覧の出力



dac /dev/hda10 の出力例

図 4-4 dac 出力例 (2 / 4) - 対象がパーティション全体の場合 - その 1



dac -Tv /dev/hda10 の出力例

図 4-5 dac 出力例 (3 / 4) - 対象がパーティション全体の場合 - その 2

```

data for dav (dac ver: 0.06)
----- check start -----
"/2"(ino: 37) is fragmented (0.01%, total: 8161/8674, frag: 0, sfrag: 1)
block sequence: x4704,Y1,x3456

----- dac result start -----
fstype  ext3
mount   unmount
f-per  0.01%
blocks 8674
sblocks -
fblocks 8161
frags  0
sfrags 1
cfiles 0
ffiles 1
depth  1
        60832  x4704,s2,i511,Y1,x3456
fbseq  x4704,Y1,x3456

```

フラグメントファイル情報(見方は図 4-5 参照)

sblocks が-(表示なし...システムブロック数は、パーティション全体確認時のみ表示) 以外は、図 4-4 と同じ

開始ブロックから順に見たブロック情報(種別+同一種別連続数をカンマで区切った文字列)を1行で出力(ブロック番号順表示)。ブロック種別の意味は図 4-4 参照。

フラグメント確認対象がファイルの場合だけ、ファイルを構成するブロックを構成順に見た場合に、どのようにフラグメントしているかを、ブロック情報(種別+同一種別連続数をカンマで区切った文字列)1行で出力(ファイル構成順表示)。ブロック種別の意味は図 4-4 参照。

dac -Tv /dev/hda10 /2 の出力例

図 4-6 dac 出力例 (4 / 4) - 対象がファイルの場合

4.3. dav 使用方法

dav は、パーティション全体 / 任意のディレクトリ下の全ファイル / 単一ファイルのいずれかのフラグメンテーション状態を取得し、GUI 表示する。また、dav 読み込み可能形式で dac 出力を行い、それをファイルに保存しておけば、いつでも dac 実行時のフラグメンテーション状況を確認することができる。

dav コマンドの説明を表 4-3 に示す。

表 4-3 dav コマンド説明

コマンド	dav	
書式	dav [-hs] [devname [path]]	
引き数	devname	デバイス名または、dac 出力ファイル名。dac 出力ファイルの場合は、dav 読み込み可能形式で出力されている必要がある。
	path	パーティション devname 内のディレクトリまたはファイル名。先頭に / (スラッシュ) が必要。path が指定されない場合は、パーティション全体のフラグメント状況を表示する。dac 出力ファイルを読み込ませる場合は、path 指定は無視される。
オプション	h	コマンドのヘルプを表示する
	s	マウントパーティション中のフラグメント状況を確認する際に、fbksinfo モジュールを使用しない。fbksinfo については「4.4 fbksinfo モジュールについて」参照

dav 実行手順を下記に示す。

- (1). スーパーユーザでログイン
- (2). fbksinfo モジュールをロード (一度ロードすれば、次回の dac / dav 実行時は不要)
/sbin/modprobe fbksinfo
- (3). dav 実行
dav (必要に応じてオプションおよび引数を指定)
ヘルプオプション (-h) 以外であれば、ここから GUI 画面が表示される。

dav の GUI 画面の各パーツの概要をに示す。

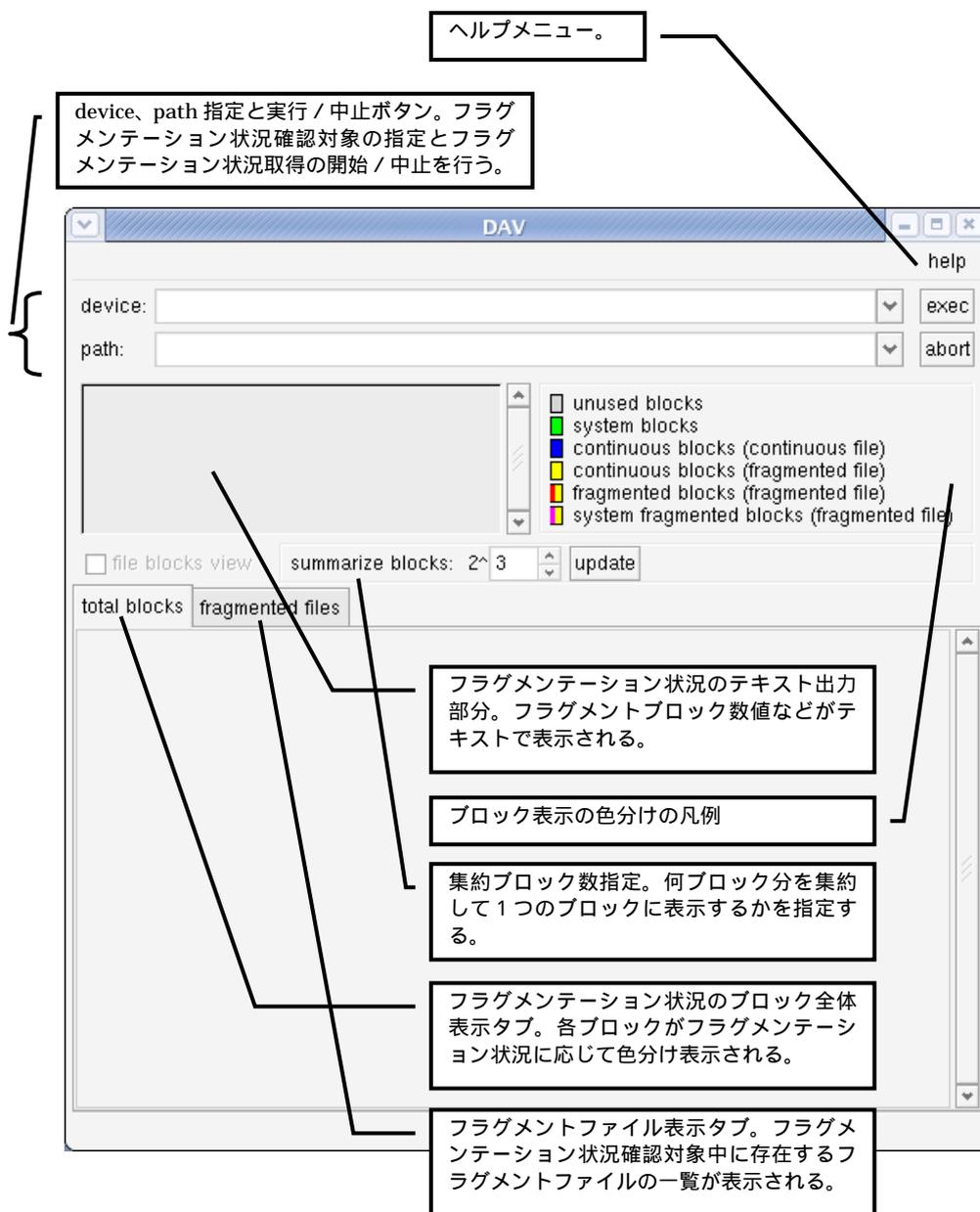


図 4-7 dav GUI 各パーツの概要

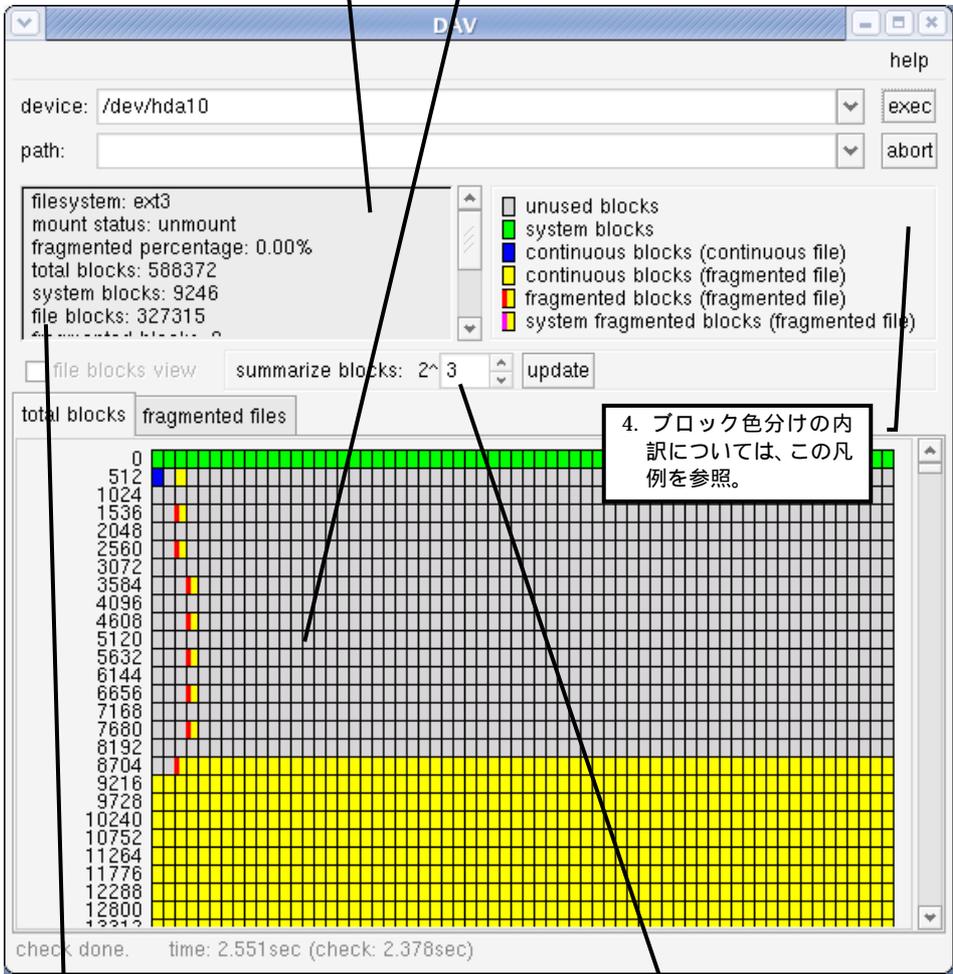
dav の GUI 画面での操作手順を図 4-8 ~ 図 4-14 に示す。



図 4-8 dav 操作手順 (1 / 7)

3. フラグメント状況取得が終了すると、テキストおよび... (3'の説明に続く)

3'. 物理的なブロック番号順に、色分けしたブロックでフラグメント状況が表示される(ブロック番号順表示)。左側の数字は、ブロック番号を示す。



4. ブロック色分けの内訳については、この凡例を参照。

5. 各行の意味は、下記の通り。
- filesystem: ファイルシステム種別
 - mount status: マウント状態
 - fragmented percentage: フラグメントパーセンテージ
 - total blocks: フラグ範囲ブロック数
 - system blocks: システムブロック数
 - file blocks: ファイル使用ブロック数
 - fragmented blocks: フラグメント数 (sys-fragmented blocks 除く)
 - sys-fragmented blocks: システムブロックによるフラグメント数
 - continuous files: フラグメントなしファイル数
 - fragmented files: フラグメントありファイル数
 - max dir depth: ディレクトリ最深階層数

6. 初期状態では $2^3 (= 8)$ ブロック分が 1 ブロックに集約されて表示される。集約ブロック数を変えるには、ここに直接数値を入力するか、すぐ右隣の / ボタンを押して値を変更してから、右側の update ボタンを押す。ブロック集約の優先度は、凡例の上から下になるほど高い。(例えば、青色と黄色を集約すると黄色になる)

図 4-9 dav 操作手順 (2 / 7)

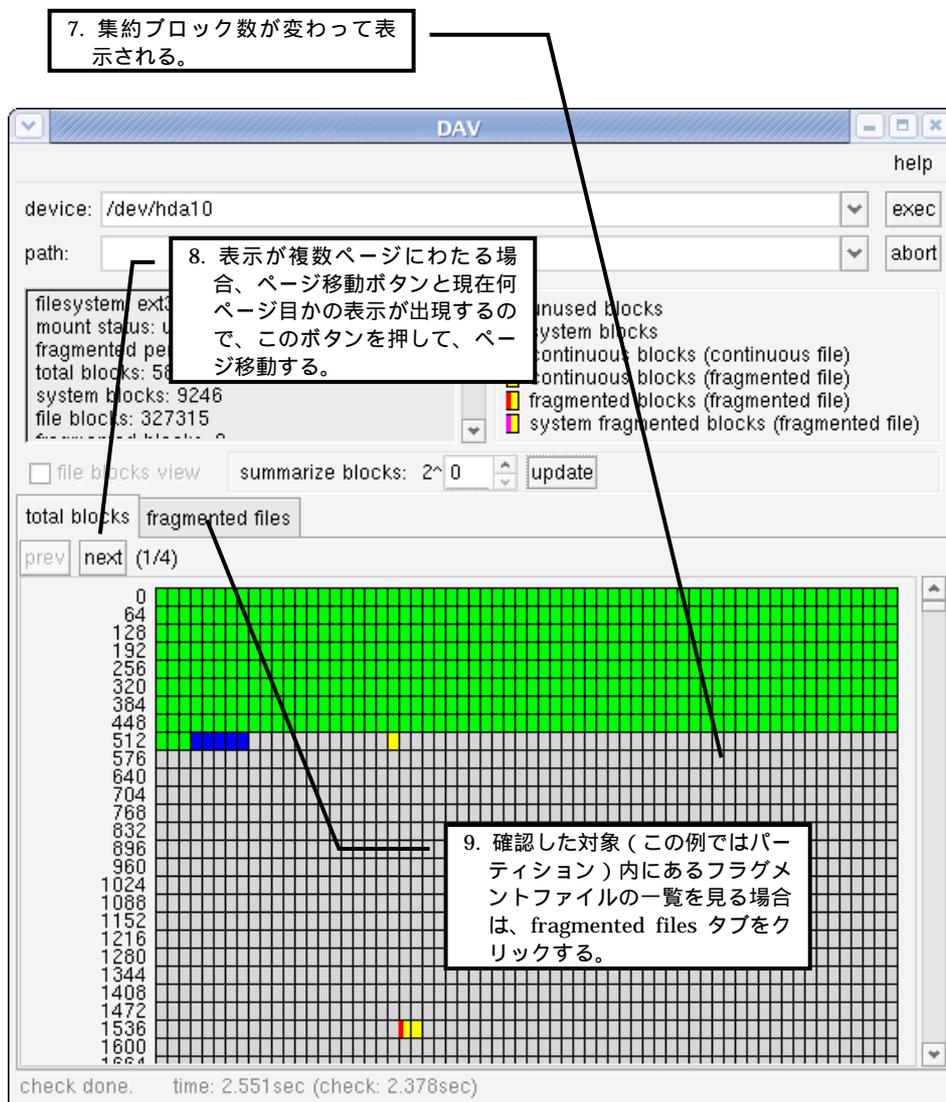


図 4-10 dav 操作手順 (3 / 7)

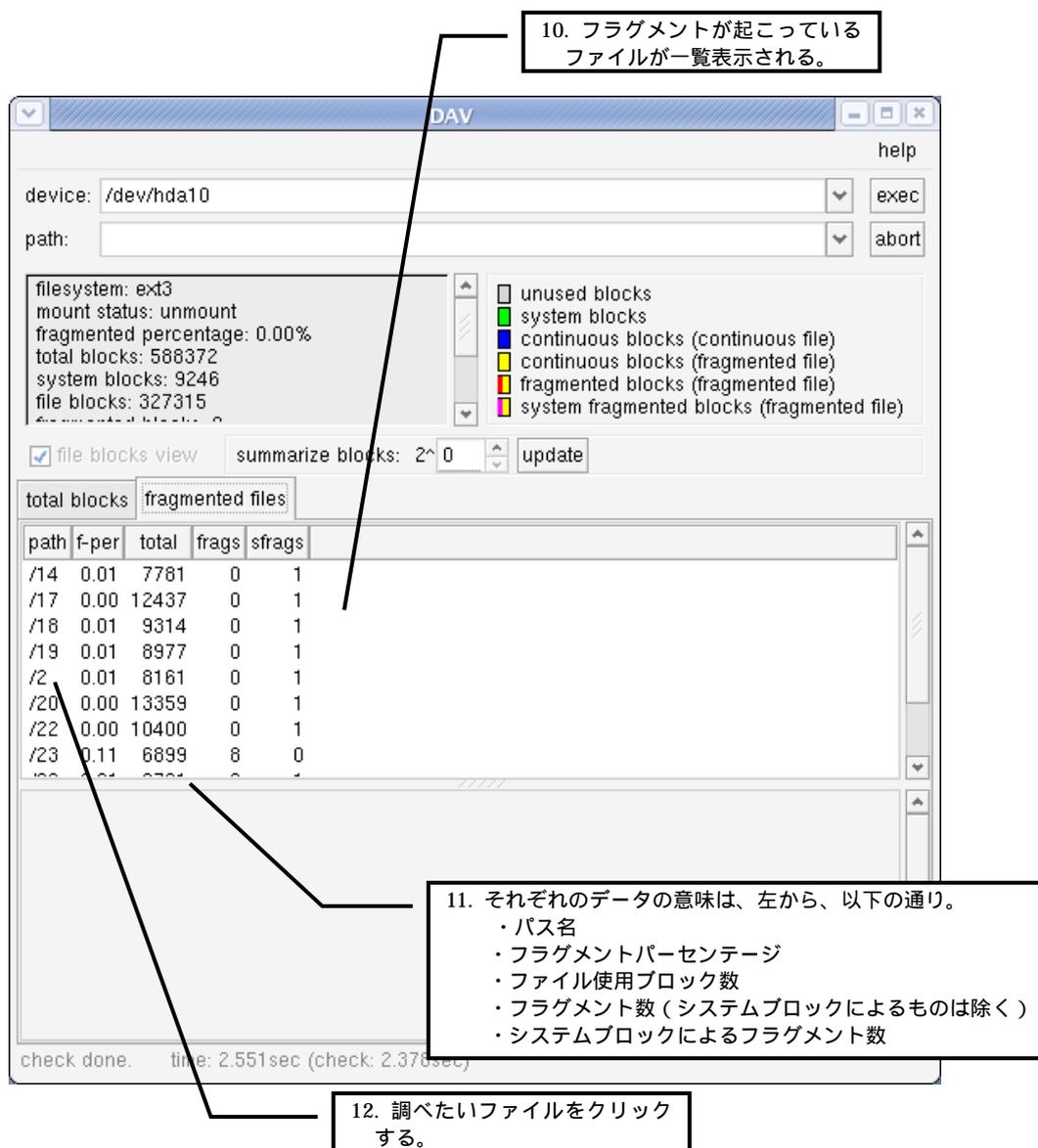
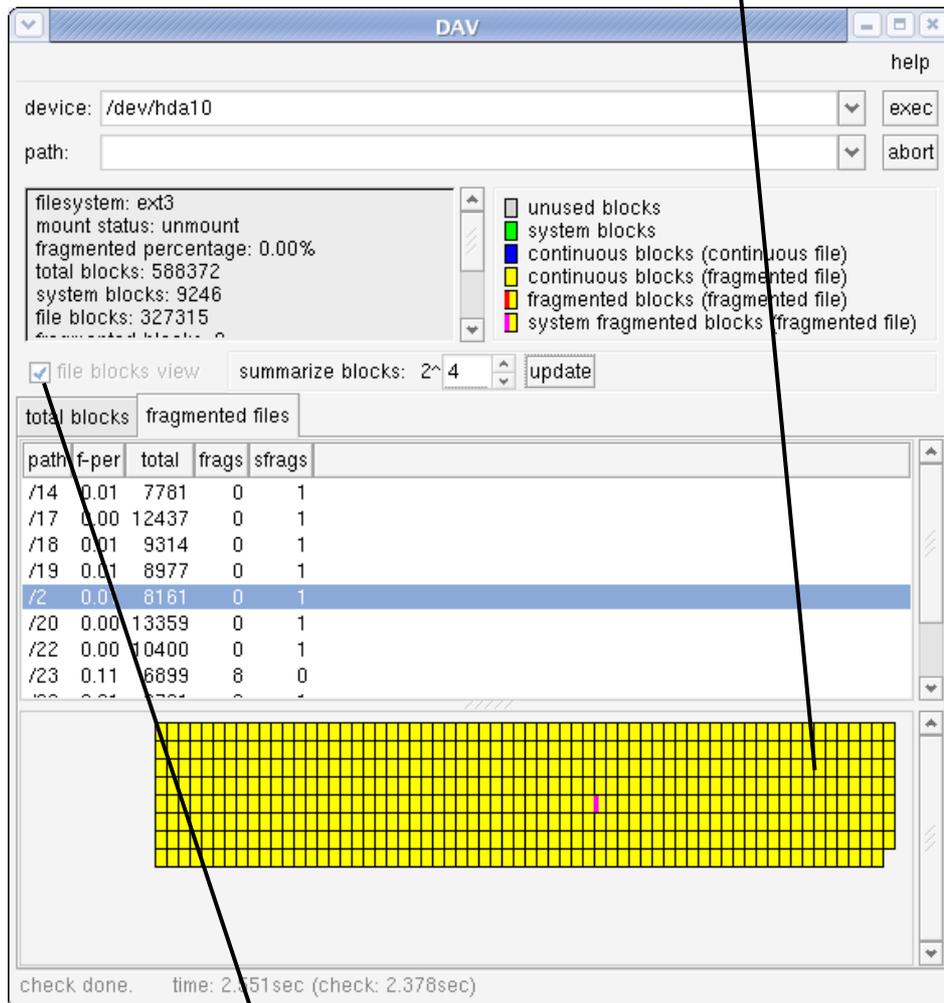


図 4-11 dav 操作手順 (4 / 7)

13. 選択されたファイルについて、ファイルを構成するブロックだけを構成順に見た場合に、どのようにフラグメントしているかを表示する（ファイル構成順表示）。ファイル構成順表示の場合は、ブロック番号順ではないため、左側にブロック番号が表示されない。



14. このチェックは、ブロック番号順 / ファイル構成順の2つの表示を切り替えるものであり、ファイル構成順表示の場合に、チェックマークが付く。fragment files タブ側では、ファイル構成順表示のみ可能である。

図 4-12 dav 操作手順 (5 / 7)

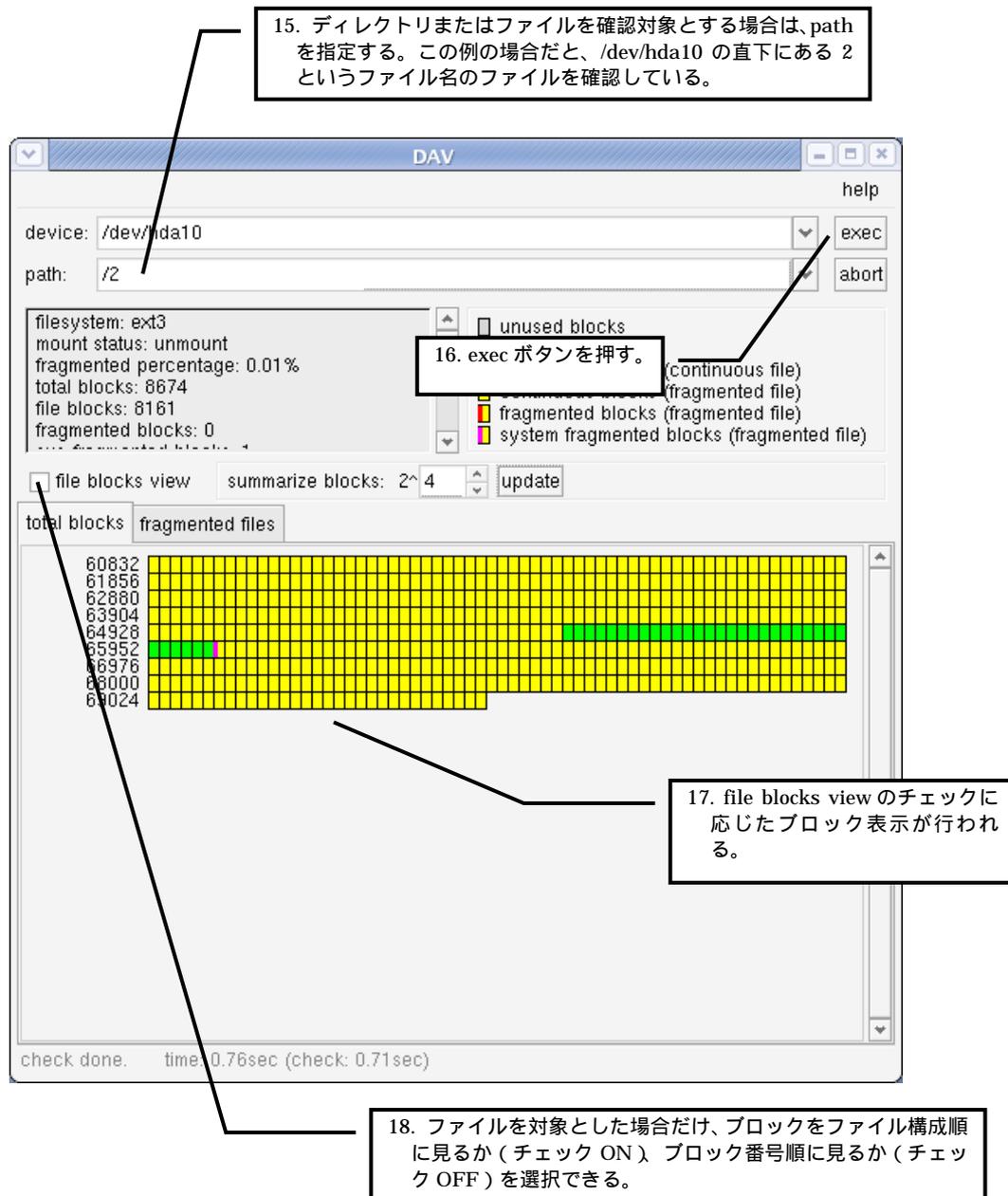
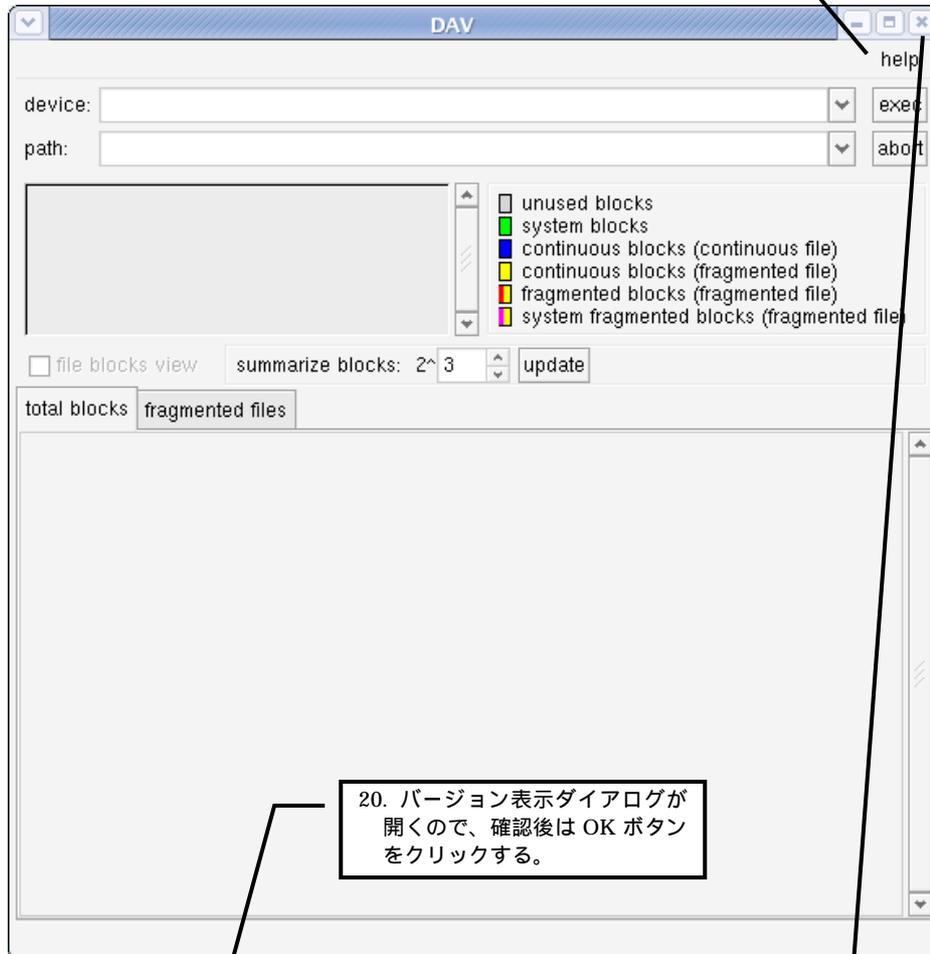


図 4-13 dav 操作手順 (6 / 7)

19. バージョンを確認するには、メニューの help version とクリックする。



20. バージョン表示ダイアログが開くので、確認後は OK ボタンをクリックする。



21. 終了する場合は、x ボタンをクリックする。

図 4-14 dav 操作手順 (7 / 7)

4.4. fbksinfo モジュールについて

DAV は確認対象がアンマウント状態のパーティションの場合、ブロックデバイスを直接読み込んでファイルブロック情報を取得する。マウント状態のパーティションの場合も同じ処理を行うと、マウントパーティション上のファイルが変更された場合、その変更がディスク I/O されるまではその変更を読み出せないため、タイミングによってはファイル変更前の情報を表示してしまう問題がある。この問題を回避するため、カーネルからファイルブロック情報を直接取得するモジュールが fbksinfo モジュールである。

DAV は、fbksinfo モジュールがロードされている状態で、かつ対象がマウント状態であれば、自動的にこのモジュールを使用する。システム負荷の軽減などの目的で、このモジュールを使いたくない場合は、`dav / dac` のコマンドオプションで `-s` を指定する必要がある。

ブロックデバイスへのファイル変更 I/O の遅延以外でも、fbksinfo モジュールを使用するかどうかで DAV に見える inode が異なる場合があり、出力結果が異なる可能性がある。モジュール使用による出力結果の差分を表 4-4 に示す。

表 4-4 モジュール使用による出力結果差分（ファイル変更 I/O 遅延以外）

項番	項目	モジュール使用時	モジュール未使用時	備考
1	シンボリックリンク	使用ブロックなしと認識	使用ブロックを正しく認識	リンク先パス名が 60 文字以上の場合のみ
2	ハードリンク	一番先にチェックしたパス名で表示	一番先にチェックしたパス名で表示	チェック順がモジュールを使用するかどうかで変わる可能性があるため、表示されるパス名が異なる場合がある
3	他ファイルシステムをマウントしているディレクトリ	使用ブロックなしと認識	使用ブロックを正しく認識	
4	inode 番号 1,3~10	使用ブロックなしと認識	使用ブロックなしと認識	ext3 ジャーナルの使用ブロックも含む

【補足～商標について】

- Linux は、Linus Torvalds の米国およびその他の国における登録商標または商標である。
- MIRACLE LINUX は、日本オラクル株式会社が登録し、ミラクル・リナックス株式会社が使用権許諾を受けている登録商標である。
- その他、記載されている会社名、製品名は、各社 の登録商標または商標である。