

LinuC レベル1 技術解説セミナー  
2022/7/30 開催

# リンク、ファイル配置

主題	1.02 : ファイル・ディレクトリの操作と管理
副題	1.02.3 : ハードリンクとシンボリックリンク 1.02.4 : ファイルの配置と検索

三澤 康巨

## 三澤 康巨

- KDDI株式会社で、サービス設備のエンジニアリングをはじめ様々な業務を担当。
- 2017年10月から2年半、KDDIグループ内のサーバ研修講師。
- サーバ研修受講者の中から、200名を超える LinuC レベル1 合格者。
- 2020年3月、KDDIを定年退職。
- LinuCレベル1技術解説セミナーの講師を担当
  - 2020年7月18日 [「ブートプロセスとsystemd」](#)
  - 2021年1月23日 [「ハードディスクのレイアウトとパーティション」](#)
  - 2021年3月 6日 [「ファイルシステムの作成と管理、マウント」](#)
  - 2021年6月26日 [「テキストデータ処理」](#)
  - 2022年1月15日 [「セキュリティ管理」](#)
  - 2022年3月12日 [「暗号の利用」](#)
- その他
  - 2020年11月28日、[オープンソースカンファレンス2020 オンライン/福岡「Linuxマシンを作ってみよう ～LinuC レベル1 / レベル2 学習環境構築ガイド～」](#)の講師を担当

## ■LinuCとは

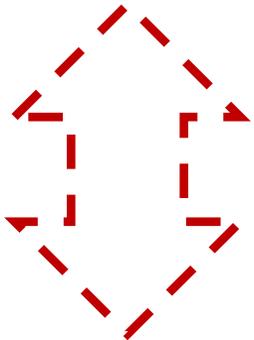
クラウド時代の即戦力エンジニアであることを証明するLinux技術者認定

- ✓現場で「今」求められている新しい技術要素に対応
  - オンプレミス／仮想化・コンテナを問わず様々な環境下でのサーバー構築
  - 他社とのコラボレーションの前提となるオープンソースへの理解
  - システムの多様化に対応できるアーキテクチャへの知見
- ✓全面的に見直した「今」身につけておくべき技術範囲を網羅
 

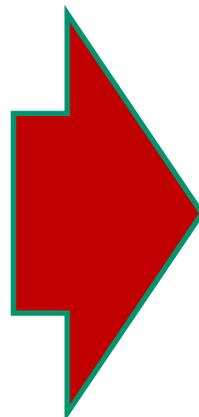
今となっては使わない技術やコマンドの削除、アップデート、新領域の取り込み
- ✓Linuxの範疇だけにとどまらない領域までカバー
 

セキュリティや監視など、ITエンジニアであれば必須の領域もカバー

AWSなどの  
パブリッククラウドを  
活用するための技術



間が  
欠けて  
いる状態



AWSなどの  
パブリッククラウドを  
活用するための技術

仮想マシン/コンテナ技術、  
クラウドセキュリティ、  
アーキテクチャ、ほか

オンプレミスの  
サーバーサイドLinux技術

オンプレミスの  
サーバーサイドLinux技術

【今まで/その他】



## リンク、ファイル配置

- 1.はじめに
- 2.ハードリンクとシンボリックリンク
- 3.ファイルの配置
- 4.ファイルの検索

# 1.はじめに

2.ハードリンクとシンボリックリンク

3.ファイルの配置

4.ファイルの検索

- 本セミナーでは、主題1.02の中から、ハードリンク、シンボリックリンク、FHS（Filesystem Hierarchy Standard）、ファイル検索について解説します。
- 学習効果を高めるため、実行例の出てくる部分では、ご自分でも実行してみることをお勧めします。
- Linuxには多数のディストリビューションが存在しますが、本セミナーの実行例では、CentOS 7 を使用します。
  - ビジネス用サーバーの多くで稼働している Red Hat Enterprise Linux 7 (RHEL7) と互換性があります。
  - RHEL7は有料ですが、CentOS 7 は無料で利用できます。
  - CentOSのバージョンは「Stream 9」までありますが、本セミナーでは、安定版であるバージョン「7」を使います。

- CentOS 7 に基づく学習環境の構築方法を、LPI-Japanのサイトでご紹介しています。

LinuC レベル1 / レベル2 Version 10.0 学習環境構築ガイド

[https://linuc.org/docs/v10/guide\\_text.pdf](https://linuc.org/docs/v10/guide_text.pdf)

- 学習環境構築ガイドでは、2種類の環境の構築方法を紹介しています。

## 【環境A】

- 用意したコンピュータの内蔵ストレージを上書きして、Linux専用コンピュータを構築します。
- WindowsやMacOS等の既存OSは使えなくなります。
- 不要になった古いPC等がある場合に、それを使ってください。

## 【環境B】

- WindowsやMacOS等の既存OSを壊すことなく、外付けSSDにLinuxをインストールします。
- これによって、既存OSとLinuxとの間を切り替えて利用することができます。
- 但し、既存OSとLinuxとを同時に利用することはできません。

※他に、VirtualBox等も紹介しています。

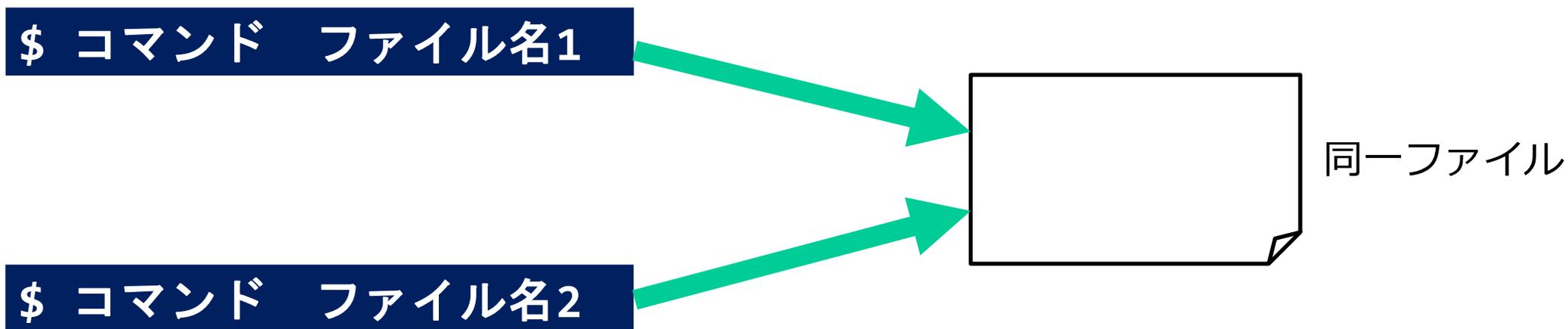
1.はじめに

**2.ハードリンクとシンボリックリンク**

3.ファイルの配置

4.ファイルの検索

- リンクとは、異なるファイル名で同一ファイルにアクセスできる仕組みです。
- Linuxには2種類のリンクがあります。
  - ハードリンク
  - シンボリックリンク
- ハードリンクとシンボリックリンクとは、同じ「リンク」でも、仕組みが全く異なります。詳しく解説していきます。



- リンクを作成するには、lnコマンドを使用します。オプションによって、生成するリンクが異なります。

## ln コマンド

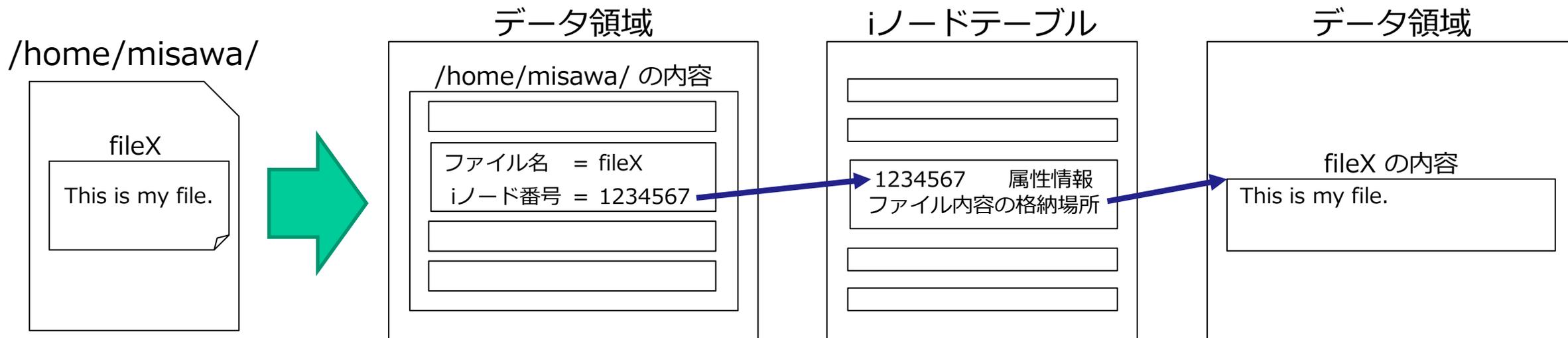
ln [オプション] リンク元ファイル名 作成するリンクファイル名

オプション	説明
なし	ハードリンクを作成
-s	シンボリックリンクを作成

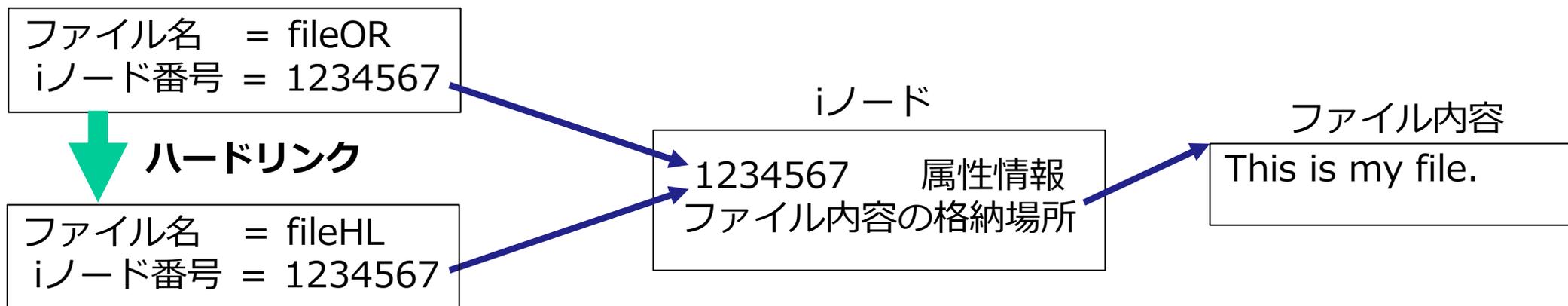
# ファイルを格納する仕組みと iノード

- ハードリンクを理解する前提として、ファイルを格納する仕組みを知ることが必要です。
- 個々のファイルは、それぞれディスク内の3カ所に分けて格納されます。

ディスク領域	格納場所	格納される内容
iノードテーブル	iノード	iノード番号 ファイルの属性情報（所有者、パーミッション、作成日付など） ファイル内容の格納場所（ポインタ）
データ領域	iノードが指す格納場所	ファイルの内容
データ領域	当該ファイルを含むディレクトリのファイル内容	ファイル名 iノード番号

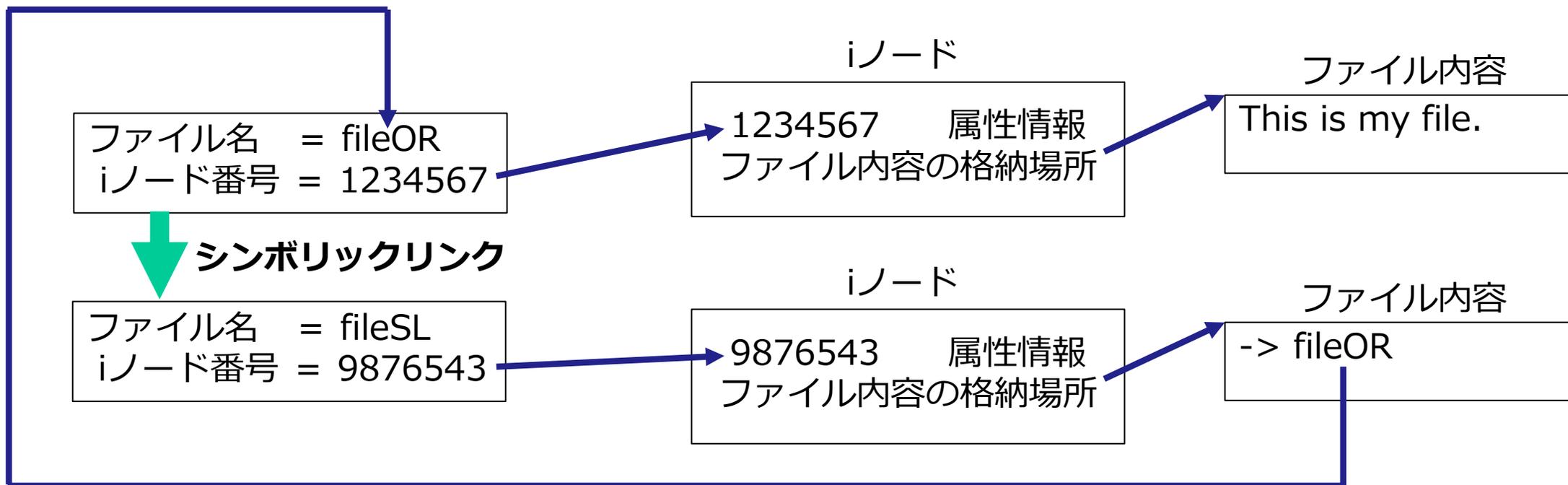


- ハードリンクは、異なるファイル名でiノードを共有する仕組みです。
- ハードリンクされたファイルは、別名の同一ファイルだと見ることができます。





- シンボリックリンクファイルは、リンク元ファイルのパスを格納した特殊なファイルです。
- 他のOSでは、ショートカットなどと呼ばれます。



## シンボリックリンクの作成と確認

```

$ ln -s fileOR fileSL      ※シンボリックリンク作成
$ ls -li file*
3964 -rw-rw-r--. 2 user1 user1 17  7月  8 18:14 fileHL
3964 -rw-rw-r--. 2 user1 user1 17  7月  8 18:14 fileOR
4359 lrwxrwxrwx. 1 user1 user1  6  7月  8 18:17 fileSL -> fileOR
  ↑  ↑           ↑
  |  |           |   ハードリンクの数は1
  |  |           |   ファイルタイプは1
  |  |           |   iノード番号が異なる
$ cat fileSL
This is my file.      ※リンク元ファイル内容を表示

```

- cpコマンドでシンボリックリンクファイルをコピーすると、リンク元ファイルのコピーが作成されます。
- シンボリックリンクファイルのコピーを作成するには、cpコマンドに -dオプションを付加します。

## シンボリックリンクのコピー

```

$ cp fileSL fileSLc1      ※リンク元ファイルのコピー作成
$ cp -d fileSL fileSLc2  ※リンクファイルのコピー作成
$ ls -l fileSL*
lrwxrwxrwx. 1 user1 user1 6 7月 8 18:17 fileSL -> fileOR
-rw-rw-r--. 1 user1 user1 17 7月 8 18:38 fileSLc1
lrwxrwxrwx. 1 user1 user1 6 7月 8 18:38 fileSLc2 -> fileOR

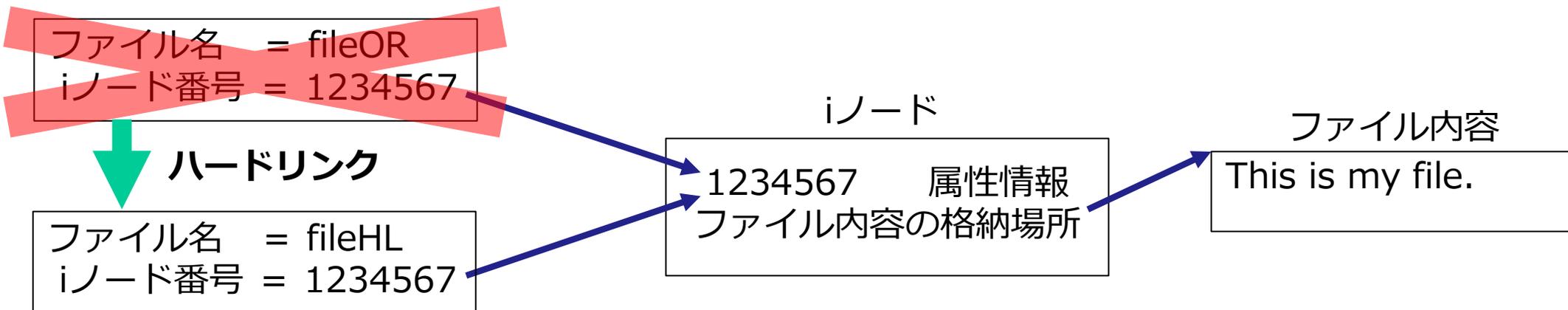
```

- リンク元ファイルを削除しても、ハードリンクファイルからファイル内容にアクセスできます。

## リンク元ファイルの削除とハードリンク

```

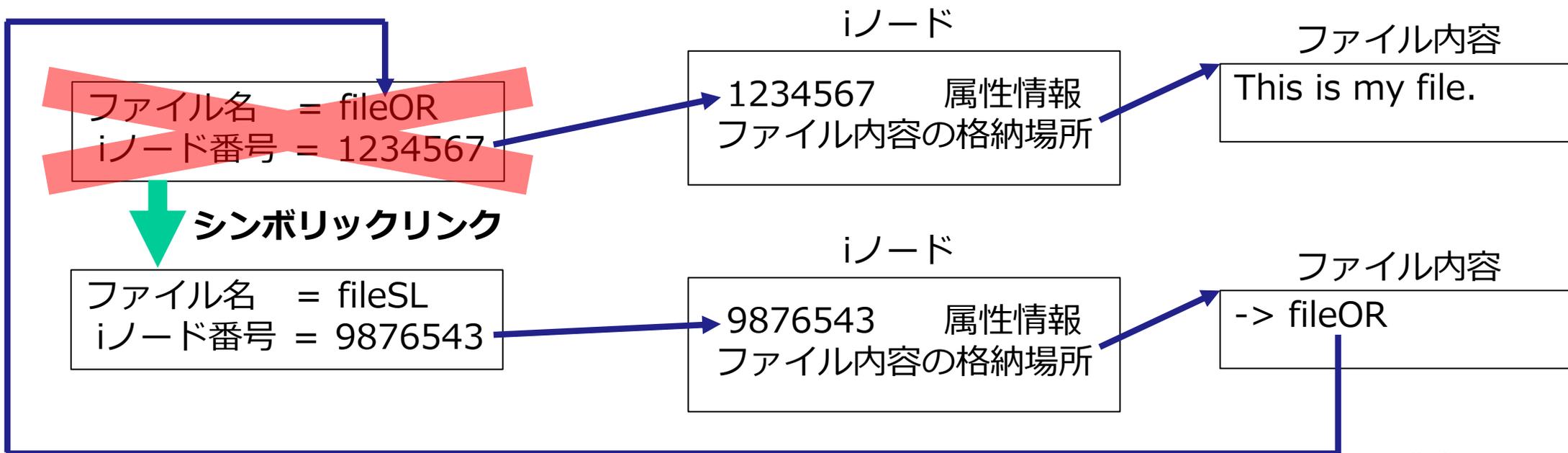
$ rm fileOR      ※リンク元ファイルを削除
$ cat fileHL
This is my file.  ※アクセス可能
$ ls -li fileHL
3964 -rw-rw-r--. 1 user1 user1 17  7月  8 18:14 fileHL
                    ↑
                ハードリンクの数が減少
  
```



- リンク元ファイルが削除されると、シンボリックリンクファイルからは元ファイルの内容にアクセスできなくなります。

## リンク元ファイルの削除とシンボリックリンク

```
$ cat fileSL
cat: fileSL: そのようなファイルやディレクトリはありません ※アクセスできない
$ ls -li fileSL
4359 lrwxrwxrwx. 1 user1 user1 6 7月 8 18:17 fileSL -> fileOR
```



- ディレクトリのハードリンクを作成することはできません。
- ディレクトリのシンボリックリンクは作成できます。

## ディレクトリのリンク作成

```

$ mkdir dirOR
$ touch dirOR/fileX
$ ln dirOR dirHL
ln: `dirOR': ディレクトリに対するハードリンクは許可されていません      ※ハードリンクを作成できない
$ ln -s dirOR dirSL
$ ls -ld dir*
drwxrwxr-x. 2 user1 user1 19  7月  8 18:26 dirOR
lrwxrwxrwx. 1 user1 user1  5  7月  8 18:27 dirSL -> dirOR      ※シンボリックリンクは作成可能
$ ls dirSL
fileX      ※dirORの内容を表示

```

- 異なるファイルシステム上にハードリンクを作成することはできません。iノードがファイルシステム毎に管理されているからです。
- シンボリックリンクは異なるファイルシステム上にも作成できます。

※ここでの「ファイルシステム」とは、ディスクにファイルを格納する仕組みのこと。詳しくは2021年3月6日開催セミナー [「ファイルシステムの作成と管理、マウント」](#) を参照。

## 異なるファイルシステム上へのリンク作成

```
$ df -h
ファイルシス      サイズ  使用  残り  使用%  マウント位置
. . .
/dev/mapper/centos-root    50G   21G   30G   42% /
/dev/sdb4                  72G    33M   72G    1% /p4
. . .
$ ln /p4/fileX fileXHL
ln: `fileXHL' から `/p4/fileX' へのハードリンクの作成に失敗しました: 無効なクロスデバイスリンク
です      ※ハードリンクを作成できない
$ ln -s /p4/fileX fileXSL
$ ls -l fileXSL
lrwxrwxrwx. 1 user1 user1 9  7月  8 18:34 fileXSL -> /p4/fileX
      ※シンボリックリンクは作成可能
```

1.はじめに

2.ハードリンクとシンボリックリンク

**3.ファイルの配置**

4.ファイルの検索

- Linuxでは共通のガイドラインに従ってファイルを配置することが推奨されています。それが FHS (Filesystem Hierarchy Standard) です。
- FHSによって、異なるホスト上でも同様の操作が可能になります。多くのディストリビューションのデフォルト設定がFHSに準拠しています。

ディレクトリ	配置するファイル
/bin	基本コマンドの実行ファイル
/sbin	システム管理用コマンドの実行ファイル
/etc	設定ファイル
/dev	デバイスファイル
/lib	ライブラリファイル、カーネルモジュール
/media	リムーバブルメディアのマウントポイント
/mnt	一時的なマウントポイント
/opt	オプションのアプリケーション
/proc	システム稼働状態を参照する仮想ファイル
/boot	システム起動用のファイル

ディレクトリ	配置するファイル
/home	一般ユーザのホームディレクトリ
/root	rootユーザのホームディレクトリ
/tmp	一時ファイル
/usr	ユーザ間で共有するファイル
/usr/bin	基本コマンドの実行ファイル
/usr/sbin	システム管理用コマンドの実行ファイル
/usr/lib	ライブラリファイル
/usr/local	ローカルなパッケージ
/usr/src	ソースコードファイル
/usr/share/doc	ドキュメント
/usr/share/man	オンラインマニュアル
/var	可変長ファイル (ログファイル、ウェブデータ、メールスプールファイルなど)
/var/log	ログファイル
/var/spool	スプールファイル (メール、プリンタなど)

## ファイル配置の例

```
$ ls -ld /bin
```

```
lrwxrwxrwx. 1 root root 7 2月 27 2021 /bin -> usr/bin
```

```
$ ls /usr/bin
```

```
... cat ... ※基本コマンド
```

```
$ ls -ld /sbin
```

```
lrwxrwxrwx. 1 root root 8 2月 27 2021 /sbin -> usr/sbin
```

```
$ ls /usr/sbin
```

```
... useradd ... ※システム管理用コマンド
```

```
$ ls -ld /tmp
```

```
drwxrwxrwt. 19 root root 4096 7月 8 18:39 /tmp ※誰でも書き込み可能
```

- 1.はじめに
- 2.ハードリンクとシンボリックリンク
- 3.ファイルの配置
- 4.ファイルの検索**

- findコマンドを使って、色々な条件でファイルを検索できます。

## find コマンド

```
find [検索ディレクトリ] [オプション] [条件式] [アクション]
```

検索ディレクトリ 検索する場所のディレクトリ（省略時はカレントディレクトリ）

### オプション

#### 説明

-maxdepth 階層数 検索するディレクトリの階層数（深さ）を制限する

### 条件式

#### 説明

-name ファイル名 ファイル名/ディレクトリ名を指定

-mtime 日時 更新日時を指定

-user ユーザ名 所有者を指定

-perm パーミッション パーミッションを指定

-type ファイル種別 ファイル種別を指定  
 f : 通常ファイル d : ディレクトリ l : シンボリックリンク

-size ファイルサイズ ファイルサイズを指定

アクション	説明
-ls	検索結果ファイルの詳細情報を表示
-exec コマンド {} \;	検索結果ファイル名を引数として指定コマンドを実行

## findコマンドによる検索

```
$ find /etc -name 'hosts*'
. . .
/etc/hosts
/etc/hosts.allow
/etc/hosts.deny
. . .
$ find / -maxdepth 1 -type l -ls
74278    0 lrwxrwxrwx   1 root    root      7  2月 27   2021 /bin -> usr/bin
74282    0 lrwxrwxrwx   1 root    root      8  2月 27   2021 /sbin -> usr/sbin
   113    0 lrwxrwxrwx   1 root    root      7  2月 27   2021 /lib -> usr/lib
   115    0 lrwxrwxrwx   1 root    root      9  2月 27   2021 /lib64 -> usr/lib64
```

- locateコマンドを使うと、ファイル名検索をfindコマンドよりも高速に実行できます。指定した文字列を名前に含むファイルとディレクトリが検索されます。
- locateコマンドは、検索用に予め構成されたデータベースを利用します。このデータベースは通常、updatedbコマンドを定期実行することによって自動的に更新されます。

## locateコマンドによる検索、updatedbコマンドによるデータベース更新

```

$ touch newfile
$ locate newfile
    ※新しいファイルは検索されない
$ sudo updatedb    ※データベースを手動で更新
$ locate newfile
/home/user1/newfile
$ cat /etc/updatedb.conf    ※updatedbコマンドの設定ファイル
. . .
PRUNEPATHS = "/afs /media /mnt /net /sfs /tmp /udev /var/cache/ccache /var/lib/yum/yumdb
/var/spool/cups /var/spool/squid /var/tmp /var/lib/ceph"
    ※/tmp 等はデータベースに反映しない

```

- FHSによると、コマンドの実行ファイルは /bin 他いくつかのディレクトリに格納されています。whichコマンド、whereisコマンド、typeコマンドを使うと、格納先を検索することができます。

## which コマンド

which [オプション] コマンド

## whereis コマンド

whereis [オプション] コマンド

## type コマンド

type [オプション] コマンド

### オプション

### 説明

-t

コマンドの種別を表示

- whichコマンドは PATH変数に登録されたディレクトリ内を検索します。

## whichコマンドによる実行ファイルの検索

```
$ echo $PATH
/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin:/home/user1/.local/bin:/home/user1/bin
$ which passwd
/usr/bin/passwd
$ which zoom | xargs ls -l
lrwxrwxrwx. 1 root root 22  2月 27  2021 /usr/bin/zoom -> /opt/zoom/ZoomLauncher
```

- whereisコマンドは、実行ファイルに加えて、マニュアルの格納先を表示します。

## whereisコマンドによる実行ファイル等の検索

```
$ whereis passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man1/passwd.1.gz
/usr/share/man/man5/passwd.5.gz
$ whereis zoom
zoom: /usr/bin/zoom
```

- typeコマンドは、実行ファイルの検索に加えて、コマンドの種別を表示することができます。

## typeコマンドによる実行ファイルの検索

```

$ type passwd
passwd は /usr/bin/passwd です
$ type zoom
zoom は /usr/bin/zoom です
$ type -t passwd
file
$ type -t echo
builtin
$ type -t ls
ls は `ls --color=auto' のエイリアスです
$ alias | grep ls=
alias ls='ls --color=auto'

```

1. CentOS 7 に基づく学習環境を自分で構築してみましょう。
2. ハードリンクとシンボリックリンクとは仕組みが全く異なり、リンク元ファイル削除時などの動作が違います。
3. Linuxでは FHS (Filesystem Hierarchy Standard) に従ってファイルを配置することが推奨されています。
4. ファイルの検索やコマンド実行ファイル格納先の検索を行うコマンドが用意されています。

ご清聴ありがとうございました