

5. Linux の概念や基本操作に関する知識 I

1. 科目の概要

Linux を利用するにあたり必要となる UNIX の知識や、Linux 発展の歴史、開発モデル、ディストリビューションの特性といった話題を紹介する。さらにシェルを介した OS の操作、マルチタスク・マルチユーザの概念、システム管理方法などについて概説する。

2. 習得ポイント

本科目の学習により習得することが期待されるポイントは以下の通り。

習得ポイント	説明	シラバスの対応コマ
I-5-1. UNIXとLinuxの関係、Linux開発の歴史	オペレーティングシステム(OS)の役割と、Linuxの手本となったUNIXの基本的な概念を説明する。さらにUNIXとLinuxの関係、Linux開発の歴史について述べる。	1
I-5-2. Linuxの開発モデルと利用上の注意点	Linux開発モデルの特徴であるバザール形式による開発スタイルを紹介する。また、LinuxおよびLinuxで動作するアプリケーションの利用に関するポイントとして、ライセンスに関する事項、ビジネス事例、典型的な利用方法などを解説する。	1
I-5-3. Linuxの配布形態	Linuxディストリビューションとは何か、その定義と特徴について説明し、代表的なディストリビューションを挙げる。また各ディストリビューションの特徴や系統(有償版/無償版、サーバ向け/デスクトップ向け)を説明する。	1
I-5-4. ユーザの概念とログイン・ログアウト	ユーザの概念について触れ、ログイン、ログアウトの手順、およびBシェル(bash)を利用したUNIX/Linuxの基本操作を説明する。またシェルを介したOSとの対話の基本的な考え方について解説する。	4,5
I-5-5. ファイルシステムの木構造とその操作	ファイルシステムは木構造をしていること、各種の種類があることなど、ファイルシステムの特徴について解説する。基本的なファイル操作/ディレクトリ操作のための代表的なコマンドとその使い方について説明する。	2,3
I-5-6. オンラインマニュアルの利用方法	オンラインマニュアルとして、manコマンドやinfoコマンドの概要を説明する。また具体的な利用方法やどのような情報を得ることができるかについて解説する。	6,7,8
I-5-7. ファイルの読み書きと入出力ストリーム	ファイルを読むためのコマンドとしてcatおよびページャ(more, less)を紹介する。さらに入出力ストリームの概念に触れ、リダイレクトやパイプの概念について解説する。	6,7,8
I-5-8. エディタの利用方法	エディタを用いてテキストファイルを編集するという概念について解説する。また、viやemacsなど一般によく利用されているエディタの基本的な操作方法を説明する。	6,7,8
I-5-9. 複数の仕事・アプリを並行処理するマルチタスク	UNIX/Linuxで処理を行う際の基本となるマルチタスクの概念を説明する。シングルタスクとマルチタスクの違い、さらにシングルユーザとマルチユーザの違いについて触れ、タイムシェアリングでマルチタスクを実現していることを説明する。	4,5
I-5-10. マルチユーザとグループ、ユーザ権限	UNIX/Linuxにおけるユーザの概念を説明する。マルチユーザ環境の意義と利点を示し、その必要性について説明する。さらに各ユーザの権限について解説し、ユーザの役割、グループによる管理方法などについて触れる。	4,5

【学習ガイダンスの使い方】

- 「習得ポイント」により、当該科目で習得することが期待される概念・知識の全体像を把握する。
- 「シラバス」、「IT 知識体系との対応関係」、「OSS モデルカリキュラム固有知識」をもとに、必要に応じて、従来の IT 教育プログラム等との相違を把握した上で、具体的な講義計画を考案する。
- 習得ポイント毎の「学習の要点」と「解説」を参考にして、講義で使用する教材等を準備する。

3. IT 知識体系との対応関係

「5. Linux の概念や基本操作に関する知識 I」と IT 知識体系との対応関係は以下の通り。

科目名	基本レベル(Ⅰ)								応用レベル(Ⅱ)						
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5. Linux の概念や基本操作に関する知識	<Linux 概要>	<ファイル操作>		<ユーザの種類と管理>		<システム管理>			<ファイルシステム>	<ファイルシステム>	<データセキュリティとバックアップ>	<データセキュリティとバックアップ>	<シェルスクリプトと開発環境>	<シェルスクリプトと開発環境>	<ネットワークの基本>

[シラバス : http://www.ipa.go.jp/software/open/oss/download/Model_Curriculum_05_05.pdf]

<IT 知識体系上の関連部分>

分野	1	2	3	4	5	6	7	8	9	10	11	12	13
情報通信システム工学	IT-IAS 情報保証と情報セキュリティ IT-IAS1. 基礎的な問題	IT-IAS2. 情報セキュリティの仕組み(対策)	IT-IAS3. 運用上の問題	IT-IAS4. ポリシー	IT-IAS5. 攻撃	IT-IAS6. 情報セキュリティ分野	IT-IAS7. フォレンジック(情報保証)	IT-IAS8. 情報の状態	IT-IAS9. 情報セキュリティサービ	IT-IAS10. 脅威分析モデル	IT-IAS11. 脆弱性		
応用技術	IT-SP 社会的な視点とグローバルなコミュニケーション IT-SP1. プロフェッショナルとしての課題	IT-SP2. コンピュータの歴史	IT-SP3. コンピュータを取り巻く社会環境	IT-SP4. チームワーク	IT-SP5. 知的財産権	IT-SP6. コンピュータの法的問題	IT-SP7. 組織の中の IT	IT-SP8. プロフェッショナルとしての倫理的な問題と責任	IT-SP9. プライバシーと個人の自由				
ソフトウェアの方法と技術	IT-PP プログラミング基礎 IT-PP1. 基本プログラミング	IT-PP2. プログラミングの基本的構成要素	IT-PP3. オブジェクト指向プログラミング	IT-PP4. アルゴリズムと問題解決	IT-PP5. イベント駆動プログラミング	IT-PP6. 再帰							
	IT-PT 技術者としてのコミュニケーション IT-PT1. システム開発	IT-PT2. データ取りまてと交換	IT-PT3. 統合的コーディング	IT-PT4. スクリプトの作成	IT-PT5. ソフトウェアセキュリティのテスト	IT-PT6. 種々の開発	IT-PT7. プログラミング言語の概要						
	IT-SNE ソフトウェア工学 IT-SNE1. 歴史と概要	IT-SNE2. 歴史と概要	IT-SNE3. ソフトウェアの要求と仕様	IT-SNE4. ソフトウェアの設計	IT-SNE5. ソフトウェアのテストと検証	IT-SNE6. ソフトウェアの保守	IT-SNE7. ソフトウェアプロジェクト管理	IT-SNE8. 言語翻訳	IT-SNE9. ソフトウェアのフォールトレザランス	IT-SNE10. ソフトウェアの構成管理	IT-SNE11. ソフトウェアの標準化		
	IT-SIA システムインテグレーションとアーキテクチャ IT-SIA1. 要求仕様	IT-SIA2. 調達/手配	IT-SIA3. インテグレーション	IT-SIA4. プロジェクト管理	IT-SIA5. テストと品質保証	IT-SIA6. 組織の特性	IT-SIA7. アーキテクチャ						
システム基礎	IT-NET ネットワーク IT-NET1. ネットワークの基礎	IT-NET2. ルーティングとスワッピング	IT-NET3. 物理層	IT-NET4. セキュリティ	IT-NET5. アプリケーション分野	IT-NET6. ネットワーク管理							
	DE-NWK テレコミュニケーション DE-NWK1. 歴史と概要	DE-NWK2. 通信ネットワークのアーキテクチャ	DE-NWK3. LAN と WAN	DE-NWK4. クラウドサービス	DE-NWK5. データのセキュリティと整合性	DE-NWK6. ワイヤレスコンピューティングとモバイルコミュニケーション	DE-NWK7. データセキュリティ	DE-NWK8. 組み込み機器向けネットワーク	DE-NWK9. 通信技術とネットワーク概要	DE-NWK10. 性能評価	DE-NWK11. ネットワーク管理	DE-NWK12. 圧縮と伸張	
	IT-PI フラットフォーム技術 IT-PI1. オペレーティングシステム	IT-PI2. アプリケーションと連携	IT-PI3. コンピュータインフラストラクチャ	IT-PI4. デプロイメントソフトウェア	IT-PI5. ファームウェア	IT-PI6. ハードウェア							
	DE-OPS オペレーティングシステム DE-OPS1. 歴史と概要	DE-OPS2. 歴史と概要	DE-OPS3. スケジューリングとパフォーマンス	DE-OPS4. セキュリティと保護	DE-OPS5. ファイル管理	DE-OPS6. リアルタイム OS	DE-OPS7. OS の概要	DE-OPS8. 設計の原則	DE-OPS9. デバイスマネジメント	DE-OPS10. システム性能評価			
ハードウェア	DE-CAO コンピュータのアーキテクチャと構成 DE-CAO1. 歴史と概要	DE-CAO2. コンピュータアーキテクチャの基礎	DE-CAO3. メモリシステムの構成とアーキテクチャ	DE-CAO4. デバイスと通信	DE-CAO5. CPU アーキテクチャ	DE-CAO6. 性能・コスト評価	DE-CAO7. 分散・並列処理	DE-CAO8. コンピュータによる計算	DE-CAO9. 性能向上				
複数領域にまたがるもの	IT-ITF IT 基礎 IT-ITF1. IT の歴史的なテーマ	IT-ITF2. 組織の問題	IT-ITF3. IT の歴史	IT-ITF4. IT 分野(学科)とそれに関連する分野(学科)	IT-ITF5. 応用領域	IT-ITF6. IT 分野における数学と統計学の活用	IT-ESY6. 要件分析	IT-ESY7. 仕様設計	IT-ESY8. 構造設計	IT-ESY9. テスト	IT-ESY10. プロジェクト管理	IT-ESY11. 逆行設計(ハードウェア、ソフトウェア)	IT-ESY12. 実装
	DE-ESY 組み込みシステム DE-ESY13. リアルタイムシステム設計	DE-ESY14. 組み込みマイクロコントローラ	DE-ESY15. 組み込みプログラム	DE-ESY16. 設計手法	DE-ESY17. ツールによるサポート	DE-ESY18. ネットワーク監視システム	DE-ESY19. センサシステム	DE-ESY20. センサ技術	DE-ESY21. デバイスドライバ	DE-ESY22. メンテナンス	DE-ESY23. 専門システム	DE-ESY24. 信頼性とフォールトレザランス	

4. OSS モデルカリキュラム固有の知識

OSS モデルカリキュラム固有の知識として、Linux の具体的な基本操作がある。シェルを通して実行されるファイル操作やユーザ管理、サービス管理などについて具体的なコマンドの実行を通して習得する。

科目名	第1回	第2回	第3回	第4回	第5回	第6回	第7回	第8回	
5. Linux の概念や基本操作に関する知識 I	(1)オペレーティングシステム(OS)の役割 (2)UNIX の基本的な概念 (3)Linux について (4)Linux にまつわるライセンスと企業での利用 (5)入手方法、ディストリビューション紹介 (6)(可能であれば)授業環境の構築方法を指	(1)ログインとログアウト (2)プロンプトとシェルの基本操作(プロンプトの役割、行編集、履歴の呼び出し) (3)コマンドの入力方法について(コマンド+引数) (4)ファイル操作 (5)ヘルプ機能 (6)ディレクトリ操作 (7)ファイルの読み書き (8)リダイレクトとパイプ		(1)ユーザとグループの概念 (2)ユーザの切り替え(su) (3)ユーザ/グループ操作 (4)ファイルのアクセス権操作 (5)ユーザ管理ファイル		(1)システムの現状分析 (2)サービスの管理 (3)ソフトウェアの組み込み			

(網掛け部分は IT 知識体系で学習できる知識を示し、それ以外は OSS モデルカリキュラム固有の知識を示している)

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-1. UNIX と Linux の関係、Linux 開発の歴史	
対応する コースウェア	第1回 (Linux 概要)	

I-5-1. UNIX と Linux の関係、Linux 開発の歴史

オペレーティングシステム(OS)の役割と、Linux の手本となった UNIX の基本的な概念を説明する。さらに UNIX と Linux の関係、Linux 開発の歴史について述べる。

【学習の要点】

- * オペレーティングシステムとはコンピュータを動かす基本ソフトウェアの一種である。
- * 近年ではミドルウェアや各種アプリケーションなども含めて、オペレーティングシステムと呼ぶことがある。
- * Linux カーネルを用いて構築されたオペレーティングシステムを、一般的に UNIX 系 OS、UNIX 互換 OS 等として分類することが多いが、基本的に UNIX と Linux とは異なるものである。

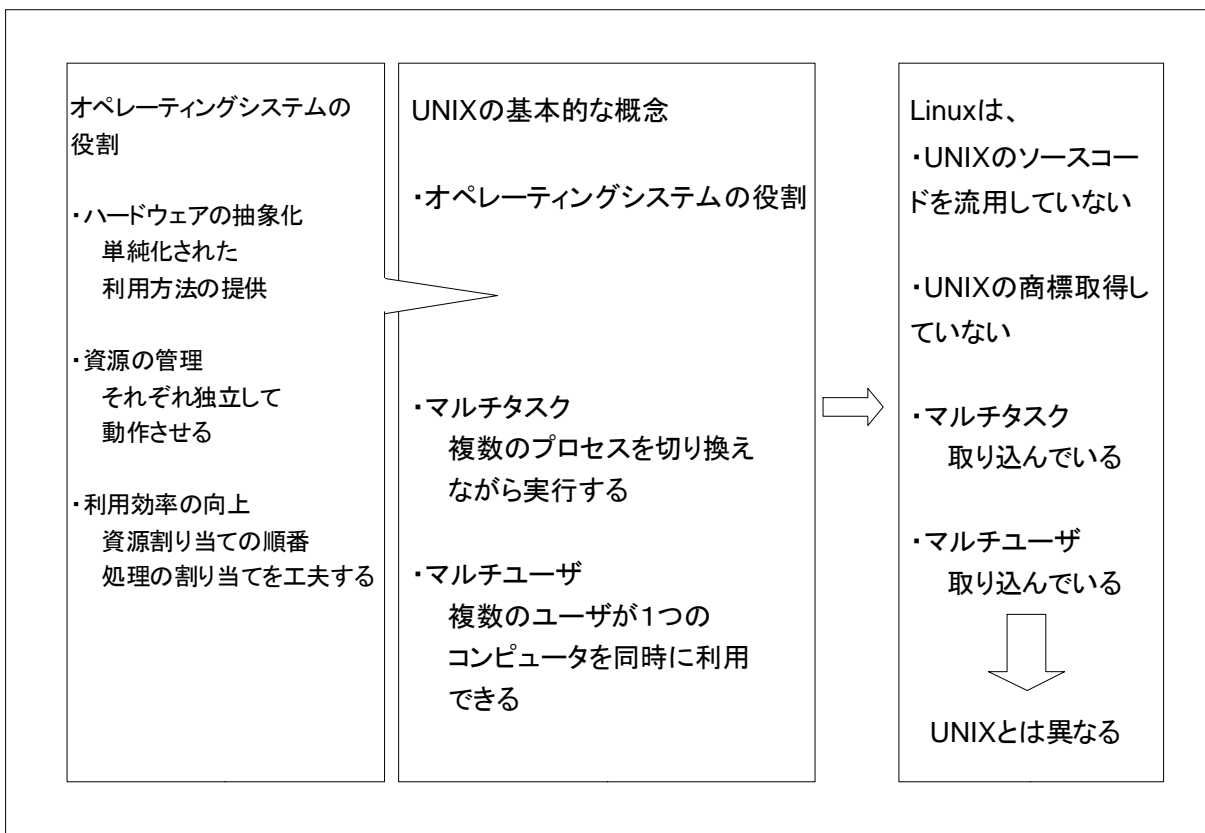


図 I-5-1. UNIX の基本的な概念と Linux との違い

【解説】

1) オペレーティングシステムの役割

オペレーティングシステムとは基本ソフトウェアの一種であり、接続されたハードウェアに対する統一的な利用環境の提供、メモリや計算資源等の資源管理と効率化といった目的を持っている。また、ミドルウェアや各種アプリケーションなどを含める場合もある。

* ハードウェア利用環境の提供

どのベンダーのハードウェアであっても、統一された利用環境を、アプリケーションに提供すること。

* 資源管理と効率化

アプリケーションプログラムを複数起動させても、利用可能な資源が確保できること。資源割り当てと処理の割り当てを工夫して、全体のスループットを向上させること。

2) UNIX と Linux の関係

UNIX は、ミニコンピュータやワークステーション用のオペレーティングシステムとして開発／採用された。特徴は、マルチタスク・マルチユーザなどの機能などを有していることである。

* マルチタスクとは

複数のタスク(プロセス)を切り替えながら、同時に実行されているように見せる機能。

* マルチユーザとは

複数のユーザが1つのコンピュータを共有すること。ユーザは、自分がそのコンピュータを専有しているように感じられる。

Linux カーネルを用いて構築されたオペレーティングシステムを、UNIX 系 OS、UNIX 互換 OS 等として分類することがある。これはマルチタスクやマルチユーザなどを含めた多くの機能を Linux も採用しているからだと思われる。ただし、UNIX の商標を取得していないことや、既存の UNIX のソースコードは一切流用していないことなどから、基本的に UNIX とは異なるものである。

3) Linux 開発の歴史

Linux カーネルは、1991 年に当時フィンランドのヘルシンキ大学在学中であったリーナス・トーバルズ氏が個人で開発をはじめたものである。最初はアセンブリ言語で記述されたターミナルエミュレータ程度のものであったが、その後は、さらに優れた Minix を作るという目標のもとにさまざまな機能が取り込まれ、拡張が繰り返された。2007 年時点で、カーネル部のソースコードは 500 万行を超えているとされる。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-2. Linux の開発モデルと利用上の注意点	
対応する コースウェア	第1回 (Linux 概要)	

I-5-2. Linux の開発モデルと利用上の注意点

Linux 開発モデルの特徴であるバザール形式による開発スタイルを紹介する。また、Linux および Linux で動作するアプリケーションの利用に関するポイントとして、ライセンスに関する事項、ビジネス事例、典型的な利用方法などを解説する。

【学習の要点】

- * バザール方式とは、複数の参加者が同時に開発をすすめる手法で、伽藍方式と対比して論じられることが多い。
- * すべての利用者は、GPL により Linux を自由に利用することが可能となっている。
- * Linux の年間成長率は 20%程度と言われており、今後ますます導入実績も増加となるだろう。

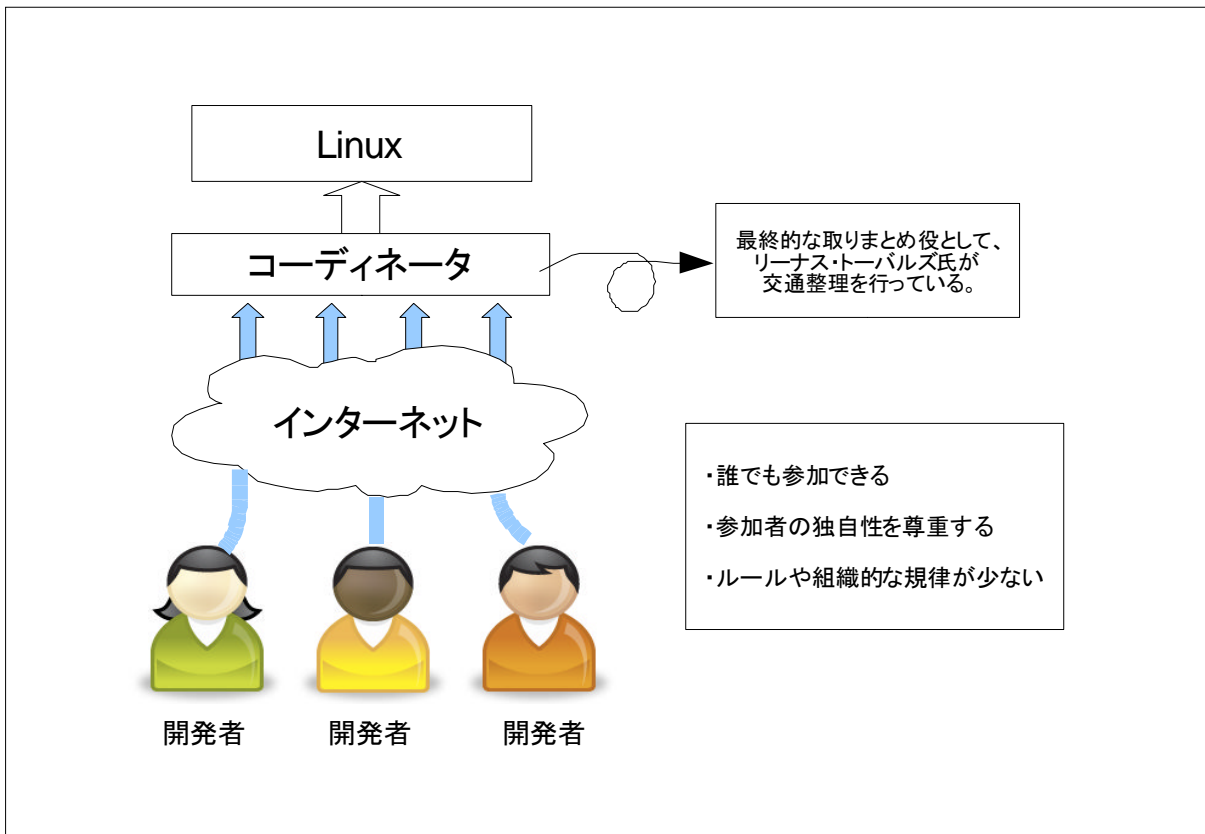


図 I-5-2. バザール方式とは

【解説】

1) Linux の成功

Linux の成功は、不特定多数の多くのプログラマが開発に参加するバザール方式と呼ばれる開発手法を採用したことによるものが多い。

* バザール方式とは

バザール方式とは、複数の参加者が特定の統率者なしに同時に開発すすめてゆく手法である。不特定多数の参加者によるもので、独自性が尊重される。この方法が、インターネット上での共同開発作業には有効であるということが Linux の開発を通じて示された。Linux の場合は、最終的な取りまとめ役としてリーナス・トーバルズ氏が意思決定を行いながら開発が進められている。

* 由来

バザール方式という名前は、エリック・レイモンド氏による論文「伽藍とバザール」で命名されたものである。選ばれた者により統率された開発を行なうスタイル(伽藍)と不特定多数により開発を行なうスタイル(バザール)が対比して論じられている。

2) ライセンスに関して

Linux は、GPL (GNU General Public License) の下で利用可能とされたカーネルと、GNU プロジェクトが開発・提供している各種サーバやアプリケーション、ライブラリなどを集めてその形を成している。したがって Linux を利用するのであれば、GPL ライセンスの内容を理解しておくのが望ましい。

* GPL とは

GPL は、プログラム本体やその複製物を所持している者に対して、以下のことを許諾するライセンスである。

- ソースコードを解読し、プログラムの動作を調べること
- ソースコードを改変すること、また、改変したプログラムを公開する権利
- ソースコードを複製すること、また、複製物の再頒布

3) Linux の導入状況

「Linux オープンソース白書 2006」(インプレス/矢野経済研究所 2006 年)等の各種調査によれば、企業ユーザの Linux サーバ導入率は現状4割程度である。しかし、現在でも Linux の年間成長率は 20%程度と言われており、今後ますます導入実績も増加となるだろう。

* 業種別導入状況

業種別に導入状況を調べてみると、公共機関での導入が飛びぬけて多い。公共機関としては、住民向けのサービスとして Web サービスやメールサーバの導入実績が多い。

* 導入時期

サービス業や製造業向けの Web サーバやファイルサーバとしての導入が一番早く、ここ数年になって金融機関向けの導入が本格化している。これは、アプリケーションサーバの信頼性向上と商用ミドルと Linux との組み合わせが、商用 UNIX に代わるものとして実績をあげてきていることなどが要因として考えられる。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-3. Linux の配布形態	
対応する コースウェア	第1回 (Linux 概要)	

I-5-3. Linux の配布形態

Linux ディストリビューションとは何か、その定義と特徴について説明し、代表的なディストリビューションを挙げる。また各ディストリビューションの特徴や系統(有償版/無償版、サーバ向け/デスクトップ向け)を説明する。

【学習の要点】

- * Linux とはカーネルと呼ばれるプログラムのことを示す。
- * Linux ディストリビューションとは、必要な機能が簡単にインストールできるようにパッケージングされたものである。
- * ディストリビューションには、商用版や無償版があり、用途ではサーバで利用するものからデスクトップでの利用を目的としているものがある。



図 I-5-3. Linux ディストリビューションに含まれるもの

【解説】

1) Linux ディストリビューションとは何か

本来、Linuxとはカーネルと呼ばれるプログラムのことを示す。しかしながら、カーネルだけではオペレーティングシステムとして機能しない。ハードウェアを利用するための様々なドライバやアプリケーションも必要となってくる。ディストリビューションとは、必要な機能を簡単にインストールできるようにパッケージングされたものである。ディストリビューションには一般に以下が含まれる。

- * インストーラ
カーネルやアプリケーションなどを、簡単にインストールさせるプログラム。代表的なものに anaconda などがある。
- * カーネル
Linux の本体。
- * シェル
オペレータの指示を、カーネルに伝えるためのプログラム。Linux は bash シェルを採用。
- * カーネルモジュール
ハードウェアにアクセスを行うために必要となるドライバなど。
- * アプリケーション
Apache、bind、sendmail などの専用サービスを提供する。オフィススイートも含まれる。
- * 便利なツール
システムをチューニングするために必要なツール類。ディストリビュータごとに開発されている。

2) 代表的なディストリビューション

ディストリビューションにも商用のものと無償のものが存在する。商用版はディストリビューション作成者(企業)が充実したサポート等を提供している。

- * Red Hat Linux 系ディストリビューション
パッケージ管理として、RPM(Redhat Package Manager)を使っているディストリビューション。
 - Red Hat Enterprise Linux: 商用。サーバ版とワークステーション版とに分かれている。
 - Fedora Core: Redhat Linux 後継の開発プロジェクト。無償。
 - SUSE Linux: SuSE 社が提供する商用版。サーバ版が存在している。
 - Turbolinux: Turbolinux 社が提供する商用版。サーバ版が存在している。
 - MIRACLE LINUX: 商用。Oracle 対応。
 - CentOS: Red Hat Enterprise Linux のクローン。無償。
- * Debian から派生したディストリビューション
パッケージ管理システムに deb 形式を使っているディストリビューション。
 - Debian GNU/Linux: コミュニティベース。
 - Ubuntu: 無償。デスクトップに主眼を置いている。
- * Slackware をベースに作られているディストリビューション
 - Plamo Linux: Slackware を日本語化したもの。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-4. ユーザの概念とログイン・ログアウト	
対応する コースウェア	第4回 (ユーザの権限と管理) 第5回 (ユーザの権限と管理)	

I-5-4. ユーザの概念とログイン・ログアウト

ユーザの概念について触れ、ログイン、ログアウトの手順、およびBシェル(bash)を利用した UNIX / Linux の基本操作を説明する。またシェルを介在したOSとの対話の基本的な考え方について解説する。

【学習の要点】

- * 1台のコンピュータを複数の人間で利用できるように、ユーザという概念が存在する。
- * 各ユーザは、互いに独立してコンピュータを占有しているかのように利用できる。
- * ユーザはコマンドを使って、カーネルに指示を与えることが可能となる。
- * カーネルは、ユーザが入力したコマンドそのままでは処理できないため、シェルが間に介在している。

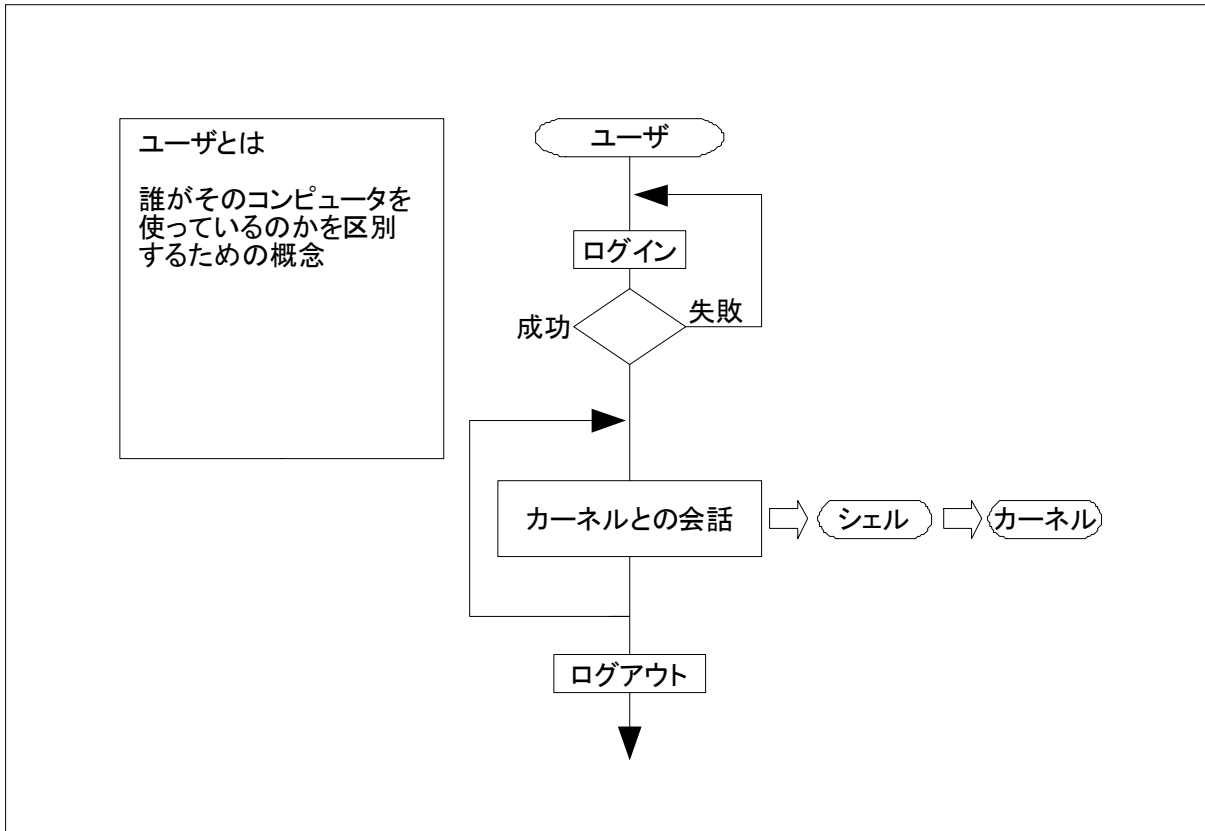


図 I-5-4. ログイン・ログアウトの概念

【解説】

1) ユーザの概念

UNIX では、当初から 1 台のコンピュータを複数の人間が利用することを想定していた。そのため一人の利用者にコンピュータを占有されることがないように、また利用者が作成したファイルを他人に書き換えられることもないように、何らかの制限を持たせなければならなくなる。そこで、誰がそのコンピュータを使っているのかを区別するためにユーザという概念が導入された。これは UNIX を手本としている Linux にも引き継がれている。

それぞれのユーザは、互いに独立してコンピュータを占有しているかのように利用できる。作業領域は /home 以下にユーザ名と同じ名前のディレクトリが用意されていて、この領域を利用する。この領域 (= ディレクトリ) を「ホームディレクトリ」と呼んでいる。

- * 利用者には、ユーザ名とパスワードといった情報が割り当てられる。
これらの情報を入手していない利用者は、当該コンピュータを使用することができない。
- * 利用者は、ユーザ名とパスワードを利用して、ログイン操作を行う必要がある。
- * ログインに成功すると、シェルを介在させてカーネルに様々な指示を送ることが可能となる。
コマンドとは、カーネルに指示を与えるための命令である。
- * 作業が終了したあとは、必ずログアウトする。

2) ログイン・ログアウトの手順

ログインを行う場合、GUI の画面が表示されていた場合とテキストモードの画面では、見た目の操作は異なるが、基本的な操作は同じである。

- * ログイン
はじめにユーザ名を入力する。続いてパスワードを入力する。正しい情報を入力するとログインに成功する。
- * ログアウト
シェルプロンプトが表示されている状態で、logout や exit などのコマンドを入力するか、[Ctrl]+[d]などのキー操作でログアウトする。

3) カーネルとの対話の基本的な考え方

カーネルに対して指示を送る場合には、コマンドと呼ばれる命令を利用する。コマンドは目的によってわけられており、使い方もそれぞれ異なっている。ただし、使い方に関して UNIX/Linux の場合は、オンラインマニュアルが豊富に用意されているためそれらを参照することができる。

- * シェルが介在する理由
カーネルには、オペレータが入力するコマンドを認識する能力はない。カーネルにオペレータの指示を理解させるためにシェルが介在しており、シェルはコマンドをカーネルが理解できる形に変換している。
- * Linux が一般的に採用しているシェルは bash
シェルにも複数の種類が存在している。Linux が一般的に採用しているシェルは bash と呼ばれるシェルである。このシェルには、TAB キーを利用したコマンド名やファイル名などの補完機能などの便利な機能が用意されている。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-5. ファイルシステムの木構造とその操作	
対応する コースウェア	第2回 (ファイル操作) 第3回 (ファイル操作)	

I-5-5. ファイルシステムの木構造とその操作

ファイルシステムは木構造をしていること、各種の種類があることなど、ファイルシステムの概念と特徴について解説する。基本的なファイル操作/ディレクトリ操作のための代表的なコマンドとその使い方について説明する。

【学習の要点】

- * オペレーティングシステムごとに、ファイルシステムが提供されている。
- * ファイルシステムとは、記憶装置に記録されているデータを管理する方式のことである。
- * ファイルシステムは木構造となっており、その頂点はルートディレクトリ("/")と呼ばれる。
- * 一般ディレクトリは、"/"から枝分かれした場所にあり、この中で様々なファイルの保管を行っている。

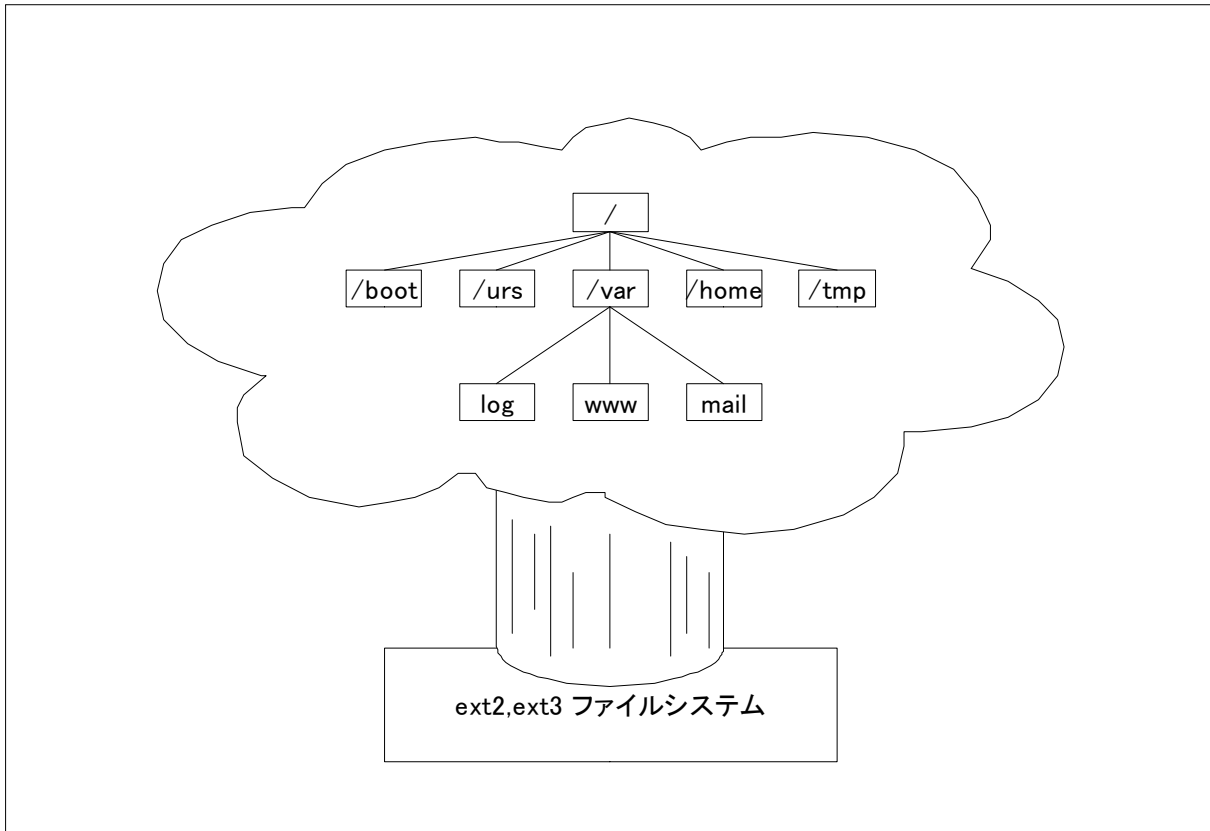


図 I-5-5. ファイルシステムの構造

【解説】

1) ファイルシステムとは

ファイルシステムとは、記憶装置に記録されているデータを管理する方式のことである。オペレーティングシステムの機能として提供されるもので、ファイルやディレクトリを作成、複製、移動、削除する方法や記録方式などが定められている。

* ext2・ext3

ext2はLinuxで広く使用されているファイルシステムの1つである。iノードと呼ばれる領域には、ファイルの所有者やサイズ、更新日、アクセス権限などの各種情報が記録されている。ext2にジャーナル機能を追加したものがext3ファイルシステムであり、現在はこのext3が主流となっている。

* FAT・VFAT・FAT32・NTFS

Windows(NTFSはWindows NT)が採用しているファイルシステム

2) ファイルシステムの構造

ファイルシステムは木構造となっている。その頂点には"/" (ルートディレクトリ)があり、他の一般ディレクトリはこの"/"ディレクトリ以下に配置される。ファイルを1つのディレクトリ下で集中して管理することは、ファイルの数が膨大に及ぶ場合には非常に難しくなってくるが、ディレクトリが複数個用意されることで用途により分類しながらファイルの管理を行うことが可能となる。

* 代表的な一般ディレクトリ

/boot、/usr、/var、/home、/tmpなどが挙げられるが、自由な名前をつけることも可能である。

* カレントディレクトリ

ユーザがUNIX/Linux上で操作を行っている時には、必ずどこかのディレクトリ内にいる。このディレクトリのことをカレントディレクトリと呼ぶ。

* ホームディレクトリ

ユーザがログインに成功した際、最初にいるディレクトリをホームディレクトリと呼ぶ。

3) ファイル・ディレクトリ操作のためのコマンド

* 代表的なファイル操作のためのコマンド

- cp :ファイルのコピーを行う
- mv :ファイルの移動を行う
- cat :ファイルの内容を参照する
- vi :ファイルの内容を編集する

* 代表的なディレクトリ操作のためのコマンド

- cd :カレントディレクトリを変更(移動)する
- mkdir:新たにディレクトリを作成する
- ls :ディレクトリ内にあるファイルなどを表示する
- pwd :カレントディレクトリ名を表示する

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-6. オンラインマニュアルの利用方法	
対応する コースウェア	第6回 (システム管理) 第7回 (システム管理) 第8回 (システム管理)	

I-5-6. オンラインマニュアルの利用方法

オンラインマニュアルとして、man コマンドや info コマンドの概要を説明する。また具体的な利用方法やどのような情報を得ることができるかについて解説する。

【学習の要点】

- * UNIX/Linux の特徴の一つに、オンラインマニュアルが充実していることがある。
- * man コマンドや info コマンドを利用すれば、引数の指定方法やオプションの指定方法などの確認を行うことが可能となる。
- * オンラインマニュアルは、一般的に日本語への翻訳が遅れており、原文で理解する必要がある。

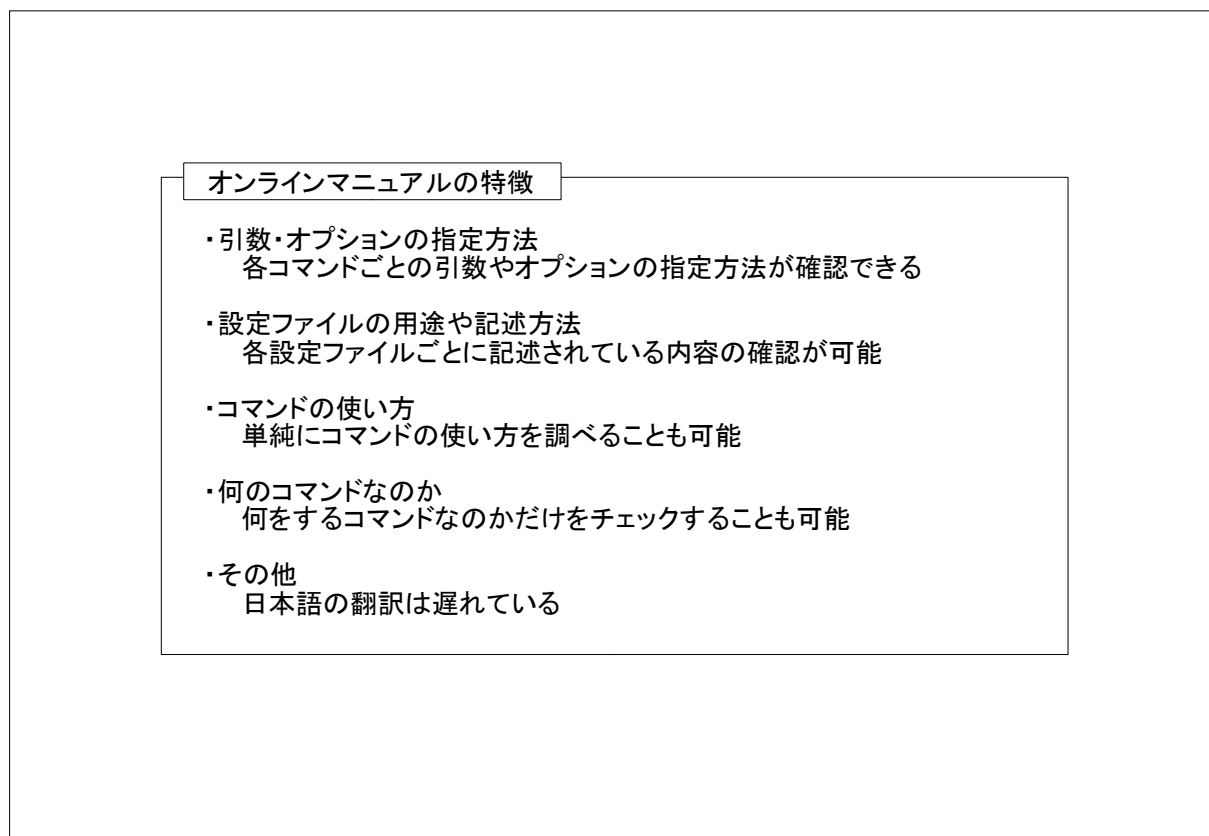


図 I-5-6. オンラインマニュアルの特徴

【解説】

1) オンラインマニュアルの特徴

UNIX/Linux の大きな特徴として、オンラインマニュアルが充実していることが挙げられる。オペレータの記憶が曖昧な部分を補う使い方であれば、十分な有効利用が可能となるはずである。

* 引数／オプションの指定方法の確認

man コマンドや info コマンドを利用すれば、引数の指定方法やオプションの指定方法などの確認を行うことが可能となる。

* 設定ファイルの用途や記述方法の確認

既存のファイルの用途や、記述されている内容の意味などの確認も可能となっている。

* コマンドの使い方であれば--help オプション

man コマンドや info コマンドなどは、様々な情報を得ることができるが、単純に個々のコマンドに関して引数の指定方法だけをチェックしたいような場合には、--help オプションなどもある。

* 何のコマンドなのかを調べたい場合には whatis コマンド

何をやるコマンドなのかをチェックしただけであれば、whatis コマンドなどもある。これは man コマンドにおける先頭行のみを表示するコマンドである。

* 注意事項

オンラインマニュアルとしては非常に充実しているが、日本語への翻訳は遅れていると言わざるをえない。日本語のマニュアルに疑問を感じた場合は、英語版のマニュアルを参照すべきである。

2) man コマンドの利用方法

man コマンドはオンラインマニュアルページを整形し表示を行う。ただし、一画面では収まらないだけの情報が表示されるので、矢印キーなどを利用しながら、必要な部分を検索し参照する必要がある。

* 利用方法

man [オプション] [セクション番号] コマンド名 or ファイル名

* セクション番号とは

man コマンドで表示されるマニュアルは、いくつかのセクションに別けられており、すべての man ページはいずれかのセクションに収められている。セクションの構成は以下の通り。

1: ユーザコマンド	4: 特別なファイル	7: その他
2: システムコール	5: ファイルフォーマット	8: 管理コマンド
3: ライブラリコール	6: ゲーム	

3) info コマンドの利用方法

man ページはクイックリファレンスとして使われ、info コマンドはより詳細な内容を紹介するドキュメントである。

* 利用方法

info [コマンド名 or ファイル名]

* Web サイトのように、別のページへのリンクも用意されている。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-7. ファイルの読み書きと入出力ストリーム	
対応する コースウェア	第6回 (システム管理) 第7回 (システム管理) 第8回 (システム管理)	

I-5-7. ファイルの読み書きと入出力ストリーム

ファイルを読むためのコマンドとして cat およびページャ (more, less) を紹介する。さらに入出力ストリームの概念に触れ、リダイレクトやパイプの概念について解説する。

【学習の要点】

- * ファイルの内容を参照するために、cat や more, less などのコマンドがある。
- * 個々のコマンドには入出力ストリームと呼ばれる3種類の I/O チャンネルが存在する。
- * コマンドを実行した結果として、何らかの情報が出力されるコマンドは、その結果を出力ストリームに出力する。
- * リダイレクトやパイプなどの機能を利用すると、デフォルトの入出力機器以外の機器からの情報入力や、結果出力が可能となる。

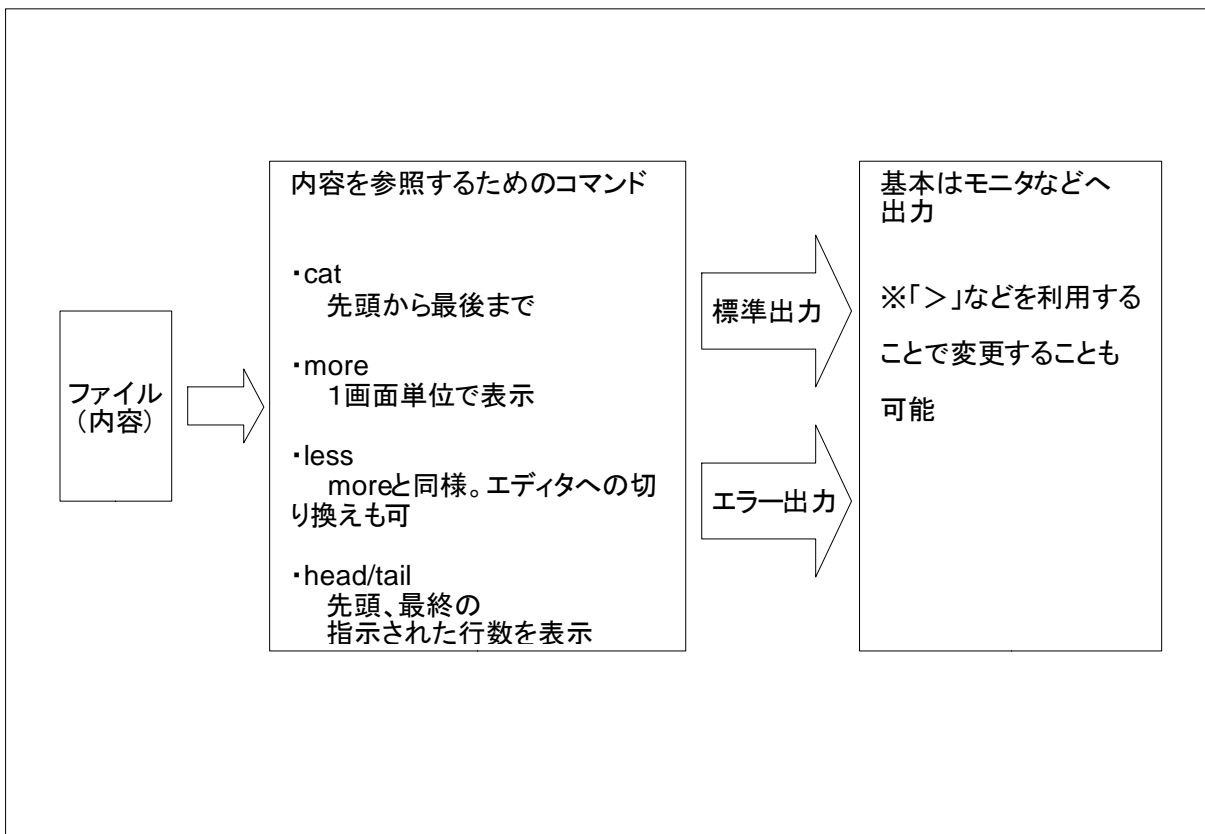


図 I-5-7. ファイルの読み書きと入出力ストリーム

【解説】

1) ファイルの内容を参照するためのコマンド

ファイルの内容を参照するためのコマンドには、`cat` や `more`、`less` などのコマンドが存在するが、それぞれの特徴を理解して使用することが重要である。

* `cat` コマンド

指定したファイルの内容を先頭行から最終行まで、すべて標準出力(デフォルトはモニタなどの表示機器)に表示する。1画面に入りきれない情報に関しては、一度表示された後、消えてしまうため、確認することができない。

* `more` コマンド

指定したファイルの内容を先頭行から1画面単位で表示する。次の画面を参照する場合にはスペースキー、次の行を参照する場合には `Enter` キーをそれぞれ押下することで可能となる。

* `less` コマンド

`more` コマンドと同様の表示を行う。表示内容の修正を行いたい場合には、`vi` エディタへ切り替えることも可能である。

* `head` コマンド/`tail` コマンド

指定したファイルの先頭、もしくは最終の指定された行数のみを表示する。

2) 入出力ストリームとは

各コマンドは、ファイルの内容を表示する際などには、表示を行いたいファイル名を何らかの形で受け取る必要がある。`cat` コマンドなどはその引数としてファイル名を受け取っているが、個々のプログラムには入出力ストリームと呼ばれる3種類の I/O チャンネルが存在する。

* 入力ストリーム(標準入力)

デフォルトの標準入力はキーボードが指定されている。キーボードから得た情報を利用して、`cat` や `more`、`less` といったコマンドはファイルの内容などの表示を行う。

* 出力ストリーム(標準出力、エラー出力)

コマンドを実行した結果として、何らかの情報が出力されるコマンドは、その結果を出力ストリームに出力する。正常な結果は標準出力に、エラーメッセージなどはエラー出力に出力される。デフォルトの出力ストリームはモニタ画面などが指定されている。

3) 入出力ストリームの変更

リダイレクトやパイプなどの機能を利用すると、デフォルトの入出力機器以外の機器からの情報入力や、結果出力が可能となる。

* リダイレクトとは

「>」や「<」などの記号を用いる。例えば、ファイルの内容を画面に表示するのではなく、他のファイルに書き込みたい場合などには、以下のように指定する。

例:`cat /etc/resolv.conf > /etc/resolv.conf.backup`

エラーメッセージだけを、ファイルに記録しておきたい場合には、以下のように指定する。

例:`tar cvfz /dev/nst0 /etc /var 2> /tmp/errormessage.txt`

* パイプとは

コマンドの結果を、他のコマンドで処理したい場合に利用する。利用方法は以下の通り。

例:`ls /etc | wc -l`

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-8. エディタの利用方法	
対応する コースウェア	第6回 (システム管理) 第7回 (システム管理) 第8回 (システム管理)	

I-5-8. エディタの利用方法

エディタを用いてテキストファイルを編集するという概念について解説する。また、vi や emacs など一般によく利用されているエディタの基本的な操作方法を説明する。

【学習の要点】

- * オペレーティングシステムを適切に機能させるための環境設定値は、すべて一般的なテキストファイルとして保存されている。
- * vi エディタは、UNIX や Linux における標準的なエディタである。
- * emacs は、OSS として公開されているエディタである。

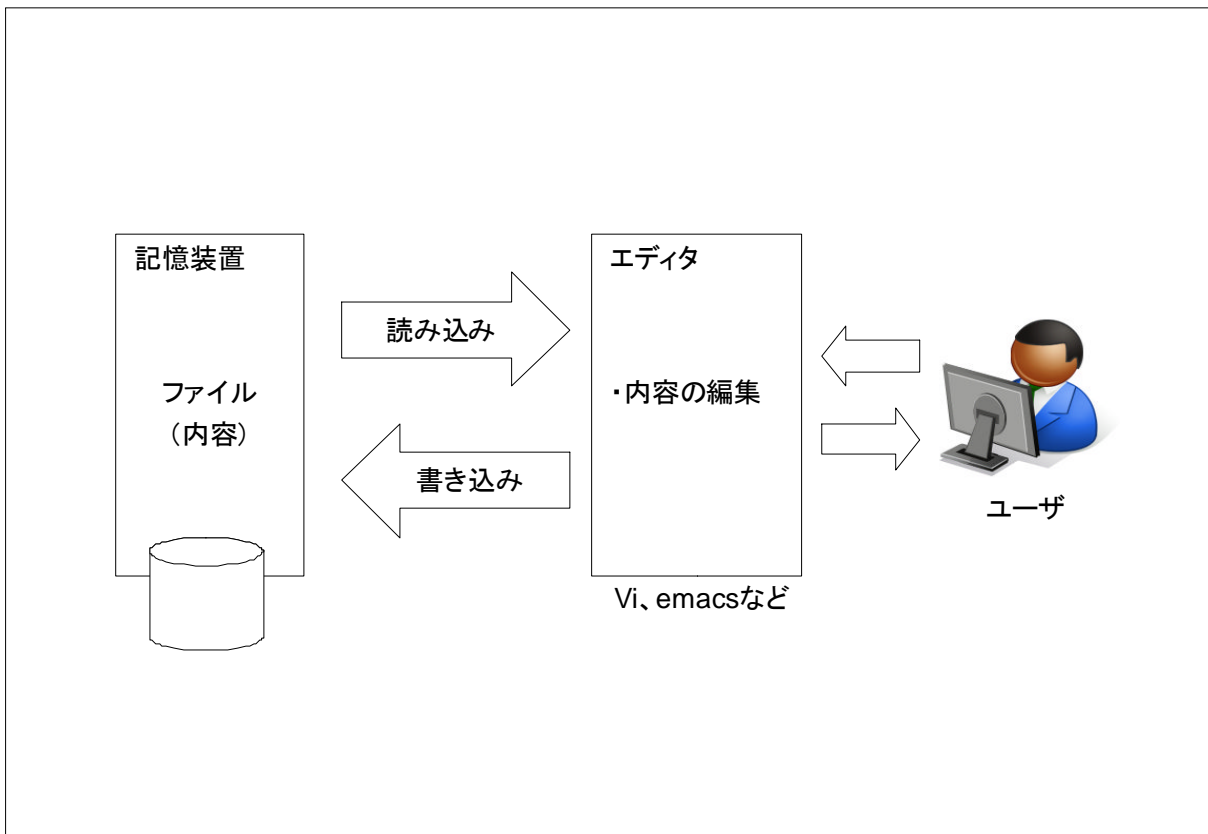


図 I-5-8. エディタの利用方法

【解説】

1) テキストファイルの編集

Linux の場合、オペレーティングシステムとして機能させるのに必要な設定値は、すべて一般的なテキストファイルの中に保存されている。そのため、機能を変更させたい場合はこの設定ファイルの内容を変更することになる。ファイルの内容を変更するためには、便利なツールやユーティリティが用意されているものもあるが、基本的にはエディタと呼ばれるプログラムを利用して変更することになる。

* エディタの種類と特徴

- vi エディタ : UNIX や Linux における標準的なエディタ。
- emacs エディタ : OSS として公開されている高機能エディタ。

2) vi エディタの基本的な操作方法

vi エディタには、コマンドモードと挿入モード、ex モードなどがあり、これらのモードを切り替えながらファイルの編集を行う。

* コマンドモード

コマンドモードでは、以下に示すコマンドが使用可能である。

- a もしくは A: 挿入モードへ切り替える。カーソルの次の文字から文字の挿入が可能。
- i もしくは I: 挿入モードへ切り替える。カーソルの場所から文字の挿入が可能。
- o もしくは O: 現在の行の前後に新しい行を追加する。
- r : 現在のカーソルの場所の文字を変更する。
- x : 現在のカーソルの場所の文字を削除する。
- : (コロン) : ex モードへ切り替える。

* ex モード

ファイルの内容を HDD に書き込んだり、vi エディタを終了させる。

- wq : ファイルの内容を書き込んでから、エディタを終了させる。
- q! : ファイルの内容は書き込まずに、エディタを終了させる。

3) emacs エディタの基本的な操作方法

Emacs には語や段落の操作やソースコードを読み易くする構文の強調機能、利用者の定義する一括編集コマンドを動かす「キーボード・マクロ」の実行のための多数の機能がある。

* 主モード

主モードと呼ばれる編集モードでは、以下に示すキー操作が可能である。

- [Ctrl]+v: 次の画面に進む。
- <ESC>+v: 前の画面に戻る。
- [Ctrl]+p, b, n, f: カーソルを移動させる。
- [Ctrl]+k: 行の削除、連続操作で完全な削除
- [Ctrl]+x [Ctrl]+f: ファイルを開く
- [Ctrl]+x [Ctrl]+s: ファイルをセーブする。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-9. 複数の仕事・アプリを並行処理するマルチタスク	
対応する コースウェア	第4回 (ユーザの権限と管理) 第5回 (ユーザの権限と管理)	

I-5-9. 複数の仕事・アプリを並行処理するマルチタスク

UNIX/Linux で処理を行う際の基本となるマルチタスクの概念を説明する。シングルタスクとマルチタスクの違い、さらにシングルユーザとマルチユーザの違いについて触れ、タイムシェアリングでマルチタスクを実現していることを説明する。

【学習の要点】

- * UNIX/Linux においては、マルチユーザ機能を有しているため複数のユーザが同時にログインすることが可能である。
- * UNIX/Linux においては、マルチタスク機能を有しているため複数のプログラムを同時に実行させることが可能である。
- * MS-DOS などでは、同時に 1 つのアプリケーションソフトしか起動できない。

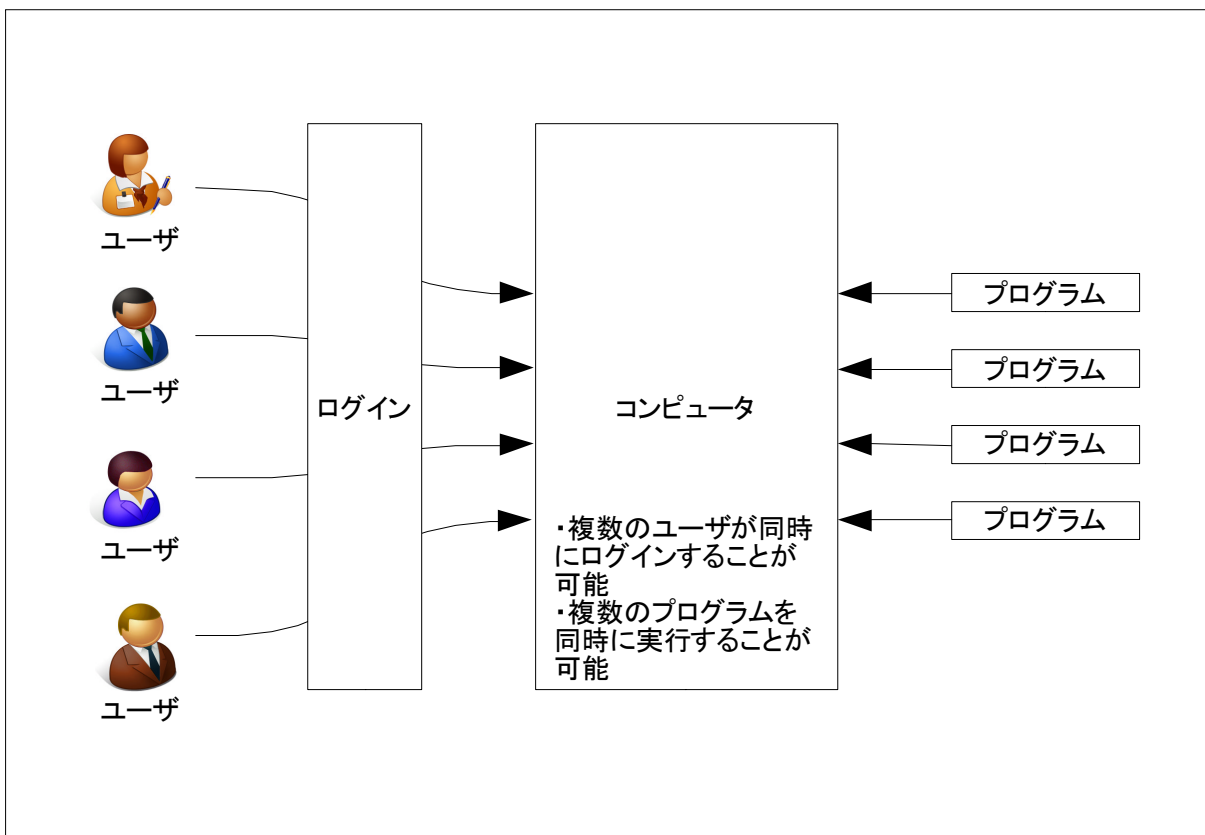


図 I-5-9. マルチユーザとマルチタスク

【解説】

1) UNIX/Linux における処理の基本

UNIX/Linux においては、複数のユーザの同時ログインが可能で、かつ複数のプログラムの同時実行も可能である。一般ユーザが一切ログインしていない状態であっても、複数個のプログラムが、外部からのリクエストに何時でも対処できるようにメモリ上に常駐して待機している。

* 背景

- UNIX が開発された当時、コンピュータは非常に高価なものであった。
- 複数ユーザで、1台のコンピュータを共有するという考え方が一般的であった。
- UNIX では、複数ユーザの利用が可能となるようにマルチユーザという機能を備えていた。

2) マルチユーザ・マルチタスクの機能

マルチユーザ・マルチタスクの機能に関して、代表的なものを以下に明記する。

* データ保存領域

マルチユーザの機能として、個々のユーザにはそのユーザのためだけのデータ保存領域が用意される。その保存領域に対して、他のユーザのアクセスが禁止されており、データ管理の安全性が保たれている。情報の共有を行うためには別途共有のための設定が必要である。

* アクセス権限

各ユーザにはユーザを識別するための ID 番号が割り振られている。ユーザが作成したファイルやディレクトリにも、作成したユーザと同じ ID 番号が割り当てられる。この ID 番号をアクセスのたびにチェックすることにより、アクセスの許可・拒否を決定している。

* CPU の処理時間分割

CPU の処理時間を非常に短い時間(単位)に分割し、複数のプログラム(タスク)に順序よく割り当てることによって、複数の処理を同時に行っているように見せている。これをタイムシェアリングと言い、この機能を活用したものがマルチタスクである。

3) シングルタスク OS

MS-DOS などのように同時に 1 つのアプリケーションソフトしか起動できない OS はシングルタスク OS と呼ばれる。UNIX や Windows、Mac OS などはマルチタスク OS である。

スキル区分	OSS モデルカリキュラムの科目	レベル
システム分野	5 Linux の概念や基本操作に関する知識 I	基本
習得ポイント	I-5-10. マルチユーザとグループ、ユーザ権限	
対応する コースウェア	第4回（ユーザの権限と管理） 第5回（ユーザの権限と管理）	

I-5-10. マルチユーザとグループ、ユーザ権限

UNIX/Linux におけるユーザの概念を説明する。マルチユーザ環境の意義と利点を示し、その必要性について説明する。さらに各ユーザの権限について解説し、ユーザの役割、グループによる管理方法などについて触れる。

【学習の要点】

- * UNIX/Linux において、そのコンピュータを使用する利用者のことをユーザと呼ぶ。
- * ログインに成功した場合、そのユーザにはそのコンピュータにアクセスを行う権限が与えられる。
- * 各ユーザには、識別のためのユーザ ID とグループを識別するためのグループ ID と呼ばれる番号が割り当てられる。

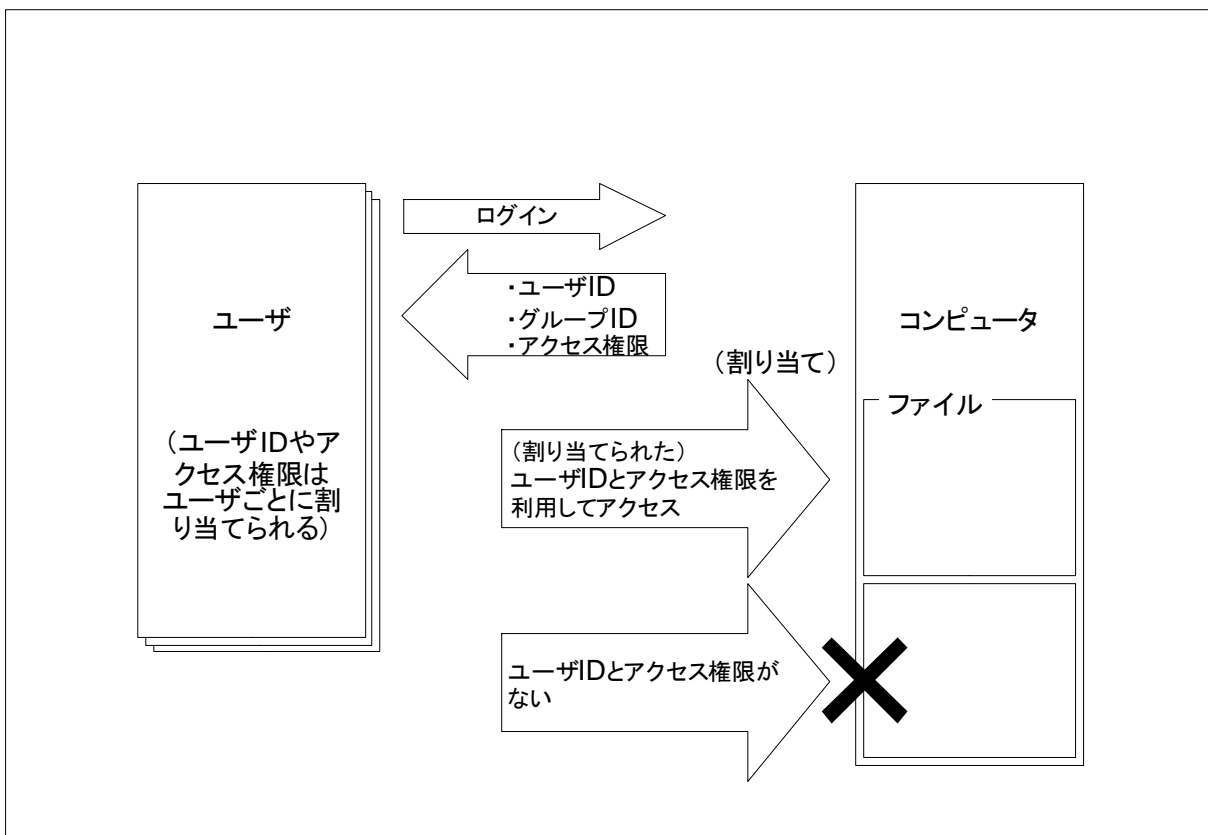


図 I-5-10. ユーザ権限

【解説】

1) ユーザとは

UNIX/Linux において、そのコンピュータを使用する利用者のことをユーザと呼ぶ。ユーザはアカウントと呼ばれるログイン ID (ログイン名やアカウント名とも呼ばれる) とパスワードの組み合わせを持ち、最初にこれらの情報を入力して承認を受けてから、実際にコンピュータの使用が可能となる。

* ユーザ認証

ログイン時には、このコンピュータを使用することを許可されているユーザかどうかの認証が行われる。ユーザは正しいログイン ID とパスワードの入力を行わなければならない。Red Hat 系ディストリビューションの場合、ログイン ID は/etc/passwd ファイルなどに保存されている。

* マルチユーザ環境

ログインが許可されるユーザは1人のみではない。複数のユーザを事前に用意することが可能であり、同時に複数のユーザがログインすることも可能である。

2) ユーザの権限

ログインに成功した場合、そのユーザにはそのコンピュータにアクセスを行う権限が与えられる。マルチユーザ機能により、他のユーザのログインも許可されることになるが、他のユーザの領域に対して、一般的な設定では、アクセスが禁止されている。

* ユーザとグループ

ログインに成功したユーザは、必ず何らかのグループに所属している。既存・新規に関わらず各ファイルや各ディレクトリにも、必ず所有者(ユーザ)と所属するグループといった情報が割り当てられている。そして、パーミッションと呼ばれる機能により、どのユーザはどのファイルやディレクトリへのアクセスが可能なのか、どのグループに所属しているユーザであれば、どのファイルやディレクトリへのアクセスが可能なのかといったアクセス権限が定められている。Red Hat 系ディストリビューションの場合、既存のグループの情報は/etc/group ファイルに保存されている。

* ユーザ ID とグループ ID

各ユーザには、ユーザを識別するためのユーザ ID と呼ばれる番号とグループを識別するためのグループ ID と呼ばれる番号が割り当てられている。オペレーティングシステムは、各ユーザの持つユーザ ID 番号とアクセス対象となるファイルやディレクトリの持つユーザ ID 番号を比較して、アクセスの許可・拒否を決定している。Red Hat 系ディストリビューションの場合、ユーザ ID やグループ ID は/etc/passwd ファイルに保存されている。

* ファイルの共有

複数のユーザ間でファイルの共有を実現したい場合には、何らかの工夫が必要となる。例えば、ファイル共有を実現したいユーザを同じグループに含めてしまう方法や、セカンダリグループを作成してこのグループを使ってファイルの共有を行うといった方法がある。

3) システム管理者 (root ユーザ)

ユーザやグループの作成や権限の変更などは、すべてシステム管理者である root ユーザのみが実行できる。