

ソフトウェア工学特論

オープンソースソフトウェア(1)

システム工学部 情報通信システム学科 大平 雅雄

自己紹介

▶ 大平 雅雄（おおひら まさお）

▶ 生年月日：1974.8.2（42歳）

▶ 出身：大阪府

▶ 学位：博士（工学）

▶ 学歴

- ▶ 1998.3 京都工芸繊維大学・電子情報工学科 卒業
- ▶ 2000.3 奈良先端科学技術大学院大学・博士前期課程 修了
- ▶ 2003.3 奈良先端科学技術大学院大学・博士後期課程 修了

▶ 職歴

- ▶ 2003.04-2003.9 科学技術振興事業団 さきがけ研究「協調と制御」領域
グループメンバー（ポスドク）
- ▶ 2003.10-2004.9 奈良先端科学技術大学院大学・EASEプロジェクト
産官学連携研究員（ポスドク）
- ▶ 2004.10-2012.3 奈良先端科学技術大学院大学・情報科学研究科
助手（2007年より助教）
- ▶ 2012.04-2014.3 和歌山大学・システム工学部・情報通信システム学科 講師
- ▶ 2014.04-現在 和歌山大学・システム工学部・情報通信システム学科 准教授

▶ 研究分野（Research Area）

- ▶ ソフトウェア工学（Software Engineering）
 - エンピリカル（実証的）アプローチ（Empirical Approach）
 - リポジトリマイニング（Mining Software Repository）
 - オープンソースソフトウェア（Open Source Software）



自己紹介：研究分野（１）

▶ オープンソースソフトウェア工学

- ▶ ソフトウェア工学の立場から、オープンソースソフトウェア（OSS）の開発／利用支援技術の確立を目指す分野



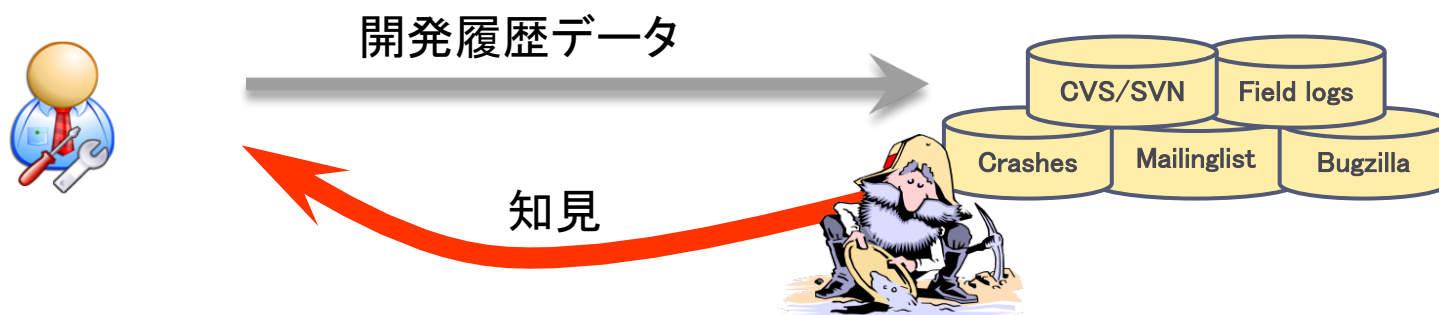
▶ ソフトウェア工学

- ▶ ソフトウェアの生産性・品質向上を支援するための技術研究領域
 - ソフトウェア開発プロセスの様々な工程が研究対象



自己紹介：研究分野（２）

- ▶ ソフトウェアリポジトリマイニング（MSR）
 - ▶ 開発データへのデータマイニング手法の応用
 - ▶ 蓄積された膨大な開発データの活用
 - 「計測」「解析」の自動化⇒PDCAサイクルの支援
 - ソフトウェア工学の中でも最近非常にホットな研究分野



- マイニングの要素技術
相関ルール分析，クラス分類，回帰分析，クラスタリング，テキストマイニング，ソーシャルネットワーク分析，など

ソフトウェア工学特論：ガイダンス（シラバス）

▶ 授業の概要・ねらい

ソフトウェア工学の基本的知識を前提とし、その特定の理論・技術について深く考えることを目的とする。（ソフトウェア工学未修の場合は教科書を紹介するので速やかにキャッチアップすること。）

各回でとりあげるテーマは多岐にわたるが、解析と合成（モデリング）、構造化・抽象化、パターンとバリエーション、トレンド予測、比較評価と確認・検証など、対象やステージ・フェーズが異なってもソフトウェア工学に通底する重要な考え方が共通して現れる。それらがさまざまな局面・文脈でどのように使われるか、何が同じで何が違うかを考察してほしい。

ソフトウェア工学知識体系（SWEBOK）との対応

知識領域	副知識領域	1998-2002 @奈良先端大	2003 @東大先端研	2004 @EASE	2005～ @奈良先端大
ソフトウェア要求	ソフトウェア要求の基礎	●	●		●
	要求プロセス	●	●		●
	要求抽出	●	●		●
	要求分析		●		●
	要求の仕様化		●		●
	要求の妥当性確認		●		●
ソフトウェア設計	実践上考慮すべきことから	●	●		●
	ソフトウェア設計の基礎				
	ソフトウェア設計における主要な問題				
	ソフトウェア構造とアーキテクチャ				
	ソフトウェア設計品質の分析と評価				
	ソフトウェア設計のための表記				
ソフトウェア構築	ソフトウェア設計戦略および手法				
	ソフトウェア構築の基礎			●	●
	ソフトウェア構築のマネジメント			●	●
	実践上考慮すべきことから			●	●
ソフトウェア試験	ソフトウェアテストの基礎				●
	テストレベル				●
	テスト技法				●
	テストに関係した計量尺度				●
ソフトウェア保守	ソフトウェア保守の基礎				●
	ソフトウェア保守における主要な課題				●
	保守プロセス				●
	保守のための技法				●
ソフトウェア構成管理	SCMプロセスのマネジメント				●
	ソフトウェア構成識別				●
	ソフトウェア構成コントロール				●
	ソフトウェア構成実態説明				●
	ソフトウェア構成監査				●
	ソフトウェアリリース・マネジメントおよび引き渡し				●
ソフトウェア工学管理	始動および適用範囲の定義			●	
	ソフトウェアプロジェクト計画			●	
	ソフトウェアプロジェクト計画実施			●	
	レビューおよび評価			●	
	終結			●	
	ソフトウェアエンジニアリング計量			●	
ソフトウェア工学プロセス	プロセスの実現および変更			●	●
	プロセス定義			●	●
	プロセス査定			●	●
	プロセスおよびプロダクト計量			●	●
ソフトウェア工学ツールと方法	ソフトウェアエンジニアリングツール	●	●	●	●
	ソフトウェアエンジニアリング手法	●	●	●	●
ソフトウェア品質	ソフトウェア品質の基礎				
	ソフトウェア品質マネジメントプロセス			●	●
	実践上考慮すべきことから			●	●

ソフトウェア工学特論：ガイダンス（シラバス）

▶ 授業計画

▶ 回 内容

- ▶ 1 オープンソースソフトウェア (1): 歴史とトレンド
- ▶ 2 オープンソースソフトウェア (2): OSS開発方法論
- ▶ 3 リポジトリマイニング (1): 概要と事例紹介
- ▶ 4 リポジトリマイニング (2): バグ予測（演習形式）
- ▶ 5 ソフトウェア開発とモデリング(1): 概要
- ▶ 6 ソフトウェア開発とモデリング(2): UMLによるモデリング
- ▶ 7 ソフトウェアのテスト(1): 概要
- ▶ 8 ソフトウェアのテスト(2): テストの技術
- ▶ 9 形式言語理論概要
- ▶ 10 記号論概要
- ▶ 11 仕様記述法の比較
- ▶ 12 XMLプロセッシングの理論
- ▶ 13 XMLプロセッシングフレームワーク
- ▶ 14 モデル駆動開発
- ▶ 15 モデル検査技術

スケジュール（大平担当分）

- ▶ 10月3日：オープンソースソフトウェア（1）
- ▶ 10月10日：休み（体育の日）
- ▶ 10月17日：オープンソースソフトウェア（2）
- ▶ 10月24日：リポジトリマイニング（1）
- ▶ 10月31日：リポジトリマイニング（2）

イントロダクション

- ▶ なぜソフトウェア工学が必要か？
- ▶ なぜソフトウェア工学でオープンソースソフトウェアを扱うのか？

なぜソフトウェア工学が必要か？ プロジェクト管理の視点

CHAOS Report

http://www.standishgroup.com/sample_research/

- In 1994
 - Only 16.2% on-time and on-budget
 - 31.1% of projects over budget
 - 52.7% of projects over schedule

On-time and On-budget

Cost/Time Overruns

Cancelled

Project Success in Japan

	Software*	Embedded Software**
Quality	52%	71%
Cost	63%	43%
Schedule	55%	36%

* Nikkei Computer, pp.36-53, No.718, Dec. 2008.

** Embedded Software Industry Survey 2006, METI Japan.

なぜソフトウェア工学が必要か？

社会経済的損失の視点

- ▶ 米国では年間595億ドル（国内総生産の0.6%）
（米国商務省国立標準技術研究所調査，2002年）
- ▶ 日本にあてはめると年間約3兆円
（ソフトウェア産業の年間売り上げのおよそ25%）



Failure on Air Traffic Control System
Cancel: 215 flights
Delay: more than 1500 flights
Affecting more than 300,000 passengers
March 1, 2003.

Failure on Auto Train Control System of JR Shinkansen.
Out of Control: 100 Shnkansen
March 22, 2005.

Failure on ATM System
Affecting 20,000 Transactions.
May 12, 2008.

Failure on Stock Trading System of Tokyo Stock Exchange
Stopped all transactions of 2520 brands in 1st and 2nd sections of TSE, Mothers.
Nov. 1, 2005

Failure on Engine Control Program on Automobiles.
Feb. 1, 2005

なぜソフトウェア工学が必要か？

<https://www.youtube.com/watch?v=xakBnXIS3Yo>

ソフトウェアを取り巻く課題とIPA/SECの取組み～安全で安心なIT社会の実現を目指して～



なぜOSSを対象とするのか？

- ▶ OSSを利用してソフトウェア開発を行う企業が増えた
 - ▶ 組み込みシステム
 - ▶ 携帯電話（Android携帯）
 - ▶ 情報家電（テレビ／ブルーレイレコーダ）， など
 - ▶ 基幹業務システム



ニューヨーク証券取引所を運営するNYSE Euronext レッドハットのソリューションを選択

～ 柔軟性、信頼性、優れたパフォーマンスを評価 ～

概要	
業種	金融取引
地域	グローバル
課題	多様化した取引プラットフォームを統合し、拡大し続ける金融取引市場の要求に迅速に対応して成果を達成できる信頼性と柔軟性を備えた、高速で低コストのプラットフォームを構築すること。
マイグレーションパス	HP UX、IBM AIX、およびSUN SolarisからRed Hat Enterprise Linuxへの移行。
ソフトウェア	Red Hat Enterprise Linux, Red Hat Network
ハードウェア	HP ProLiant DL585 4プロセッササーバ 200台、ProLiant BL 685cブレードサーバ 400台、AMDデュアルコアOpteronプロセッサ
利点	信頼性が高くセキュリティ機能と費用対効果に優れたソリューションの実装により、特定ベンダーに拘束されない柔軟性を提供して膨大な作業負荷の処理を可能にするとともに、迅速な対応による成果を実現。



メガバンクのCIOが語る基幹OSとしてのLinux

レッドハットが先週、Red Hat Enterprise Linuxの基幹システム向けサポート強化を発表した会見で、メガバンクのCIOが基幹OSとしてのLinuxについて語った。

【松岡功, ITmedia】

Linuxの基幹システムへの適用が当たり前に

レッドハットが4月11日、Red Hat Enterprise Linux (RHEL) 5および6のサポート期間を、これまでの7年から10年に延長すると発表した。

また、ミッションクリティカル分野向けの追加サポートである「Red Hat Advanced Mission-Critical Program」(AMC) をRHEL 6にも適用すること、AMCでのRHEL 5のサポート期間を13年（2020年まで）に延長することも発表した。

2012年04月16日 08時00分 更新

Google カスタム検索

特集インデックス

- ・ ソフトウェアが変えるビジネス最前線
- ・ “ビッグデータ”が経営を変える
- ・ デジタルマーケティングの時代
- ・ 2012年それぞれの「スタート」
- ・ 中小企業の活力を高めるIT活用の潮流
- ・ 現場で効くデータ活用と業務カイゼン

Special

Facebook おすすめ Twitter

ITmedia エンタープライズ

24時間以内にアカウントを停止します
—Facebookを名乗る新たな詐欺攻撃が発生
8,109人がすすめています。

Appleからの誤情報に基づく詐欺メール、新規
ユーザーは特に注意を
48人がすすめています。

最後によかったな—と考える人生 (1/3)

なぜOSSを対象とするのか？

- ▶ OSS／OSSベースのシステムの利用者（エンドユーザ）が増えた
- ▶ 政府機関／自治体等での導入
 - ▶ MS Windows/Officeの代替品（コスト削減目的）
- ▶ モバイルOSの普及（＝Android）
- ▶ Firefox OSやTizen搭載のスマホも間もなく製品化

The screenshot shows the ITpro website with a main article titled "「コスト削減と雇用創出に効果」、自治体がオープンソース/Linux" (Effect on cost reduction and job creation, local governments evaluate open source/Linux). The article discusses the adoption of OSS by local governments, mentioning the "Open Source Software Conference 2010 Government" and the use of Linux. It also mentions the "Open Source Software Conference 2010 Government" and the use of Linux. The article is dated 2011/03/01 and is written by ITpro. The article is part of a series of articles about OSS and Linux.

The screenshot shows the Yahoo! Japan website with a main article titled "2011年の脆弱なスマートフォン・ランキング - トップ12全てがAndroid" (2011's Most Vulnerable Smartphones Ranking - Top 12 all Android). The article lists the top 12 most vulnerable smartphones of 2011, all of which are Android. The list includes: 1. Samsung GALAXY Mini, 2. HTC Desire, 3. Sony Ericsson Xperia X10, 4. Sanyo Zio, 5. HTC Wildfire, 6. Samsung Epic 4G, 7. LG Optimus S, 8. Samsung GALAXY S, 9. Motorola DROID X, 10. LG Optimus One, 11. Motorola DROID 2, 12. HTC Evo 4G. The article also mentions that the list is based on a security report from The Dirty Dozen.

なぜOSSを対象とするのか？

- ▶ OSSを作る側（OSSプロジェクト）／活用する側（企業）／利用する側（エンドユーザ），それぞれが問題を抱えるようになった

OSSコミュニティで発生している問題

1. 開発リソースが不足している
2. 一部の開発者に負担が集中し過ぎている
3. 不具合管理が限界にきている
4. 有能な開発者がすぐに離脱する
5. ユーザ企業からの支援がほとんどない



安心・安全にOSSを開発・利用できる環境の実現

OSS導入の際のユーザ企業の懸念

1. 緊急時の技術的サポートが得にくい
2. 利用しているOSSがいつまで存続するかわからない
3. バグの改修や顧客からの要請対応に手間がかかる
4. 将来のバージョンアップの計画が把握しにくい
5. セキュリティホールに対するコミュニティの対応に不安

ユーザ企業



エンドユーザ



脆弱なスマホランキング
上位11機種Androidが独占!

オープンソースソフトウェア (OSS)

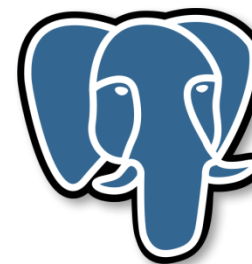
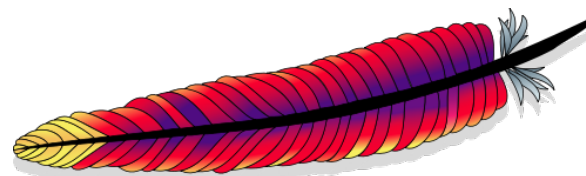
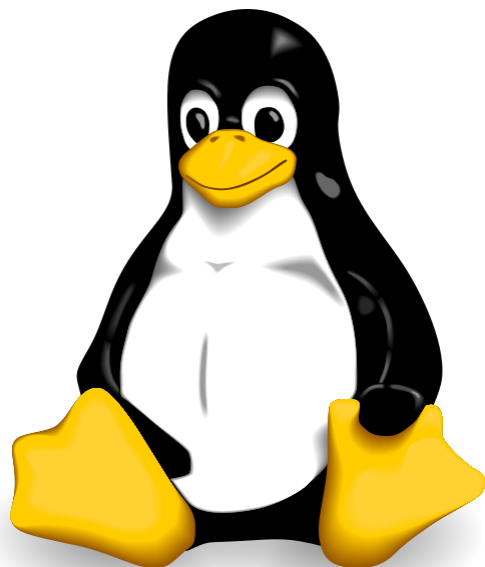
オープン？

ソース？

身近なオープンソースソフトウェア



意識せずとも利用しているOSS



LAMP = Linux + Apache + MySQL + PHP

組み込みシステムにも強いOSS

- ▶ 一般家電製品
 - ▶ 炊飯器
 - ▶ 洗濯機
 - ▶ エアコン
- ▶ デジタル家電
 - ▶ デジタルカメラ
 - ▶ テレビやハードディスクレコーダーなどのAV機器
 - ▶ PC周辺機器（プリンタ，スキャナ，ルーターなど）
 - ▶ ゲーム機
 - ▶ 携帯電話
 - ▶ カーナビ

総合 ニュース 企業とIT システム導入 経営 パソコン ケータイ eBook デジカメ AV機器 ガジェット ライフ動画 ブログ 旬ネタ

ITmedia ニュース

BPnet ビジネス PC IT テクノロジー 医療 建設・不動産 TRENDY WOMAN ショッピング ◆復興ニッポン | 転職 | NATIONAL GEOGRAPHIC | 日経電子版

ITpro

Focus BYOD 標的型攻撃 ビッグデータ ソーシャル 中国・アジア

経営 情報システム ネットワーク セキュリティ ソフト開発 ニュース 特集 解説 インタビュー コラム 事例 調査 selfup キーワード 辞書 イベント

Information System

ITpro > 情報システム > オープンソース/Linux

OSC2010.Government レポート

「コスト削減と雇用創出に効果」、自治体がオープンソースを評価

2011/03/01
高橋 信輔 ITpro
出典：日経BPカバメントテクノロジー 2010年10月1日号 p.10
(記事は執筆時の情報に基づいており、現在では異なる場合があります)

「オープンソースと政府・自治体」をテーマにしたイベントが、東京都日野市の明星大学で2010年9月10日・11日に開催された。オープンソースソフトウェア(OSS)を導入した自治体が、導入の効果や課題について講演、議論を交わした。

イベントの名称は「オープンソースカンファレンス(OSC)2010.Government」。OSS関連コミュニティが日本各地で開催しているOSCのテーマイベントとして企画された。「政府や自治体のIT予算削減要求が高まっており、その手段としてOSSが注目されている」とOSC実行委員の宮原徹氏は開催の動機を語る。

会津若松市、塩尻市、箕面市、大館市、長崎県、徳島県などが成果報告

会津若松市 情報政策課 目黒純氏は同市におけるオープンソースソフトウェアの活用を報告した。同市では標準オフィスソフトとしてオープンソースのOpenOffice.orgを採用しており、2008年から5年かけて市役所の全パソコン約840台のうち85%をMicrosoft OfficeからOpenOffice.orgに移行し、5年間で1500万円を削減する計画を進めている(関連記事)。

長野県塩尻市はRubyやLinuxで構築した図書館システムを採用した。塩尻市は同市の全市民図書館に「費用を既存システムの開発・保守に充てる」という方針で、地元のベンダーに発注することで雇用を創出

読みましたか? ~ 未読記事をご紹介します

- 【記者の眼】6月6日に始まる「World IPv6 Launch」で何が...
- 【週末スペシャル】COMPUTEX TAIPEI 2012で探る、Ultrab...
- 【記者の眼】ドコモの「ネットワーククラウド」戦略に感じるいくつかの疑問
- 【IT提案書 基礎スキル】いきなりPowerPointの図を描き始めるのはやめよう
- 【田中克己の針路IT】受託ソフト開発会社は、もう終わり!
- 【続・奥井規晶の悩むなら聞け!】「仕事に興味がない人材」を育てる方策を知りたい。自信喪失社員...

情報システムの最新記事 >>一覧

- OpenPaaSとは?
- SigCheckを使いこなす
- 頭取、社長が率先して利用
- オープンソースで安価な災害救助ラジコン飛行機、「OpenRelief」プロジェクト...
- 【COMPUTEX 2012】Windows 8タブレットやUltrabookで人気を集めるASUSブース
- 【COMPUTEX 2012】台湾MSI Thunderbolt接続の携帯ノード向け画像ユニット

アクセスランキング

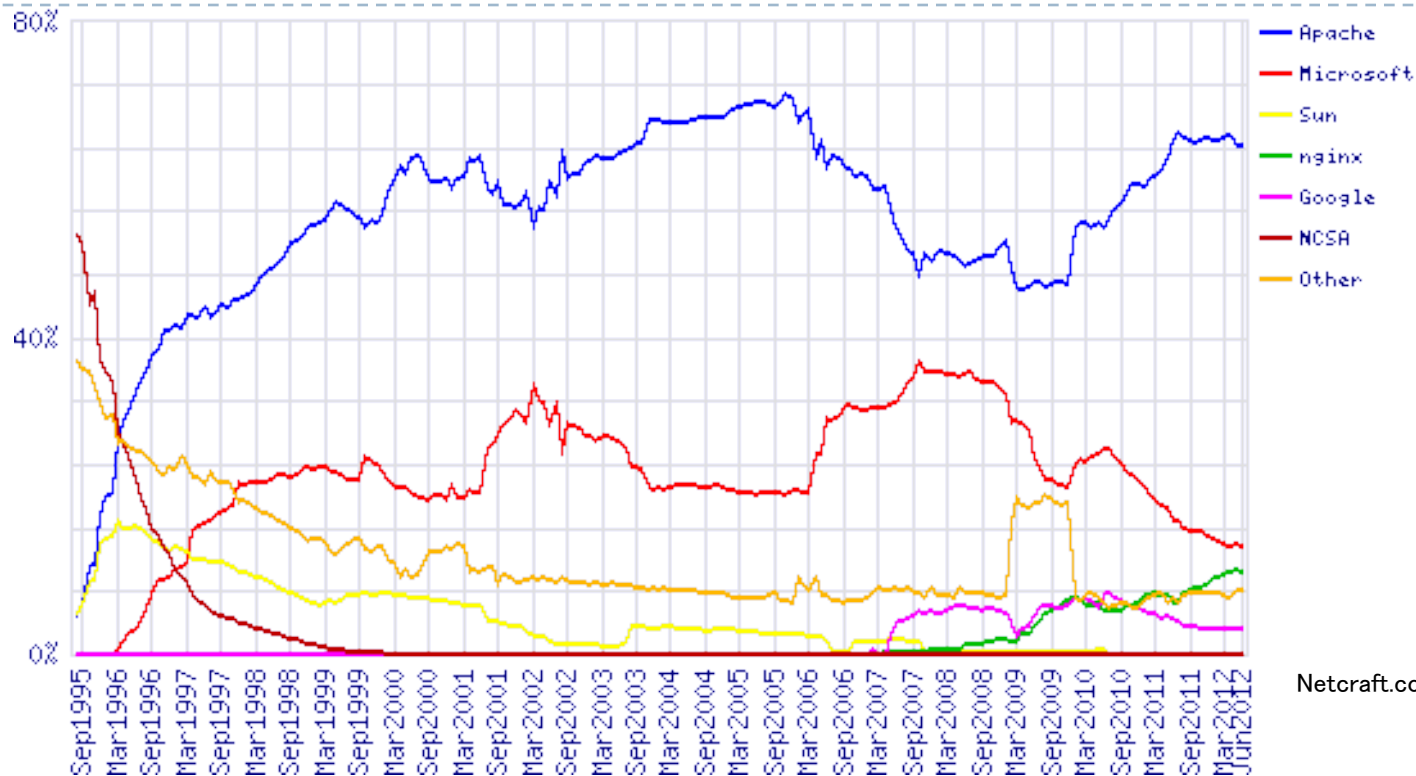
オープンソース/Linux		情報システム	
昨日	週間	昨日	週間
2012年06月09日			
1	【ニュース】マイクロソフト、Windowsの3割安でLinuxの仮想マシンサービス開始	1	【ニュース】オープンソースで安価な災害救助ラジコン飛行機、「OpenRelief」プロジェクト発足
2	【記者の眼】もっと気楽にLinuxで遊んでみ	2	

https://www.facebook.com/plugins

関連記事 from ITpro Premium

オープンソースのシステム開発では「Apache使用のWebサイト」が最多

参考： Apacheの市場シェア（2012年6月時点）



Developer	May 2012	Percent	June 2012	Percent	Change
Apache	425,631,721	64.20%	448,452,703	64.33%	0.13
Microsoft	92,406,480	13.94%	95,891,537	13.76%	-0.18
nginx	70,764,248	10.67%	72,881,755	10.46%	-0.22
Google	21,264,616	3.21%	22,464,345	3.22%	0.02

オープンソースのメリット・デメリット

▶ メリット

- ▶ 導入コストが安いこと
 - ▶ ソースコードを誰でも入手・利用できるため
- ▶ 問題の発見・解決が迅速なこと
 - ▶ ソースコードが世界中に公開されているため
- ▶ 自由度が高いこと
 - ▶ 改変が認められているため、自分好みに機能拡張したり、変更が容易することができるため
- ▶ ベンダー・ロックインを回避する手段になること
 - ▶ 本講義後半の内容に関連

オープンソースのメリット・デメリット

▶ デメリット

- ▶ サポートが得られるかどうか不透明なこと
 - ▶ 問題が発生したときに確実にサポートが得られるかどうかわからない
 - ▶ ドキュメントが十分に整備されていないこともある
 - ▶ 開発プロジェクトが突然解散したり，分裂することがある
- ▶ 結果的に運用コストが高くつく場合がある

オープンソースソフトウェア

- ▶ 定義 (The Open Source Definition by OSI)
 - ▶ 自由な再頒布ができること
 - ▶ ソースコードを入手できること
 - ▶ 派生物が存在でき、派生物に同じライセンスを適用できること
 - ▶ 差分情報の配布を認める場合には、同一性の保持を要求してもかまわない
 - ▶ 個人やグループを差別しないこと
 - ▶ 適用領域に基づいた差別をしないこと
 - ▶ 再配布において追加ライセンスを必要としないこと
 - ▶ 特定製品に依存しないこと
 - ▶ 同じ媒体で配布される他のソフトウェアを制限しないこと
 - ▶ 技術的な中立を保っていること

オープンソースソフトウェアの開発形態

- ▶ コミュニティ主導のソフトウェア開発
 - ▶ 世界中のボランティア開発者がオンライン上にコミュニティを形成し、共同でソフトウェアを開発する
 - ▶ 共同開発により得られた成果であるソフトウェア（ソースコード）は 無償で公開される

なぜこのような開発形態が定着したのか？



Questions

- ▶ 誰が作ってるのか？
- ▶ どうやって作ってるのか？
- ▶ 何のために作ってるのか？
- ▶ いつまで開発が続くのか？
- ▶ OSS利用時のサポートはあるのか？などなど

OSS開発に関する情報はまだまだ一般的ではない



OSSを活用してソフトウェア開発を行うためには、まずOSS開発に関する知識を十分に身につけておかなければならない

本講義のねらい

- ▶ 情報通信システム開発におけるOSSの重要性を理解し、OSS開発に関する基礎知識を習得すること
 - ▶ オープンソース開発の歴史
 - ▶ オープンソース開発の方法論
 - ▶ オープンソース開発・利用支援に関する研究
- ▶ 注意点
 - ▶ オープンソースの利用を勧めたり、オープンソース開発にまつわる思想を植え付けるための講義ではありません
 - ▶ OSSの前提・基礎知識を学んでもらい、将来、エンジニアとしてOSSテクノロジーを利用する可能性の高い諸君に心構えをしてもらうことが目的です

オープンソースソフトウェア開発の歴史

1950～1970年代（1）

▶ メインフレーム(大型汎用コンピュータ) の時代



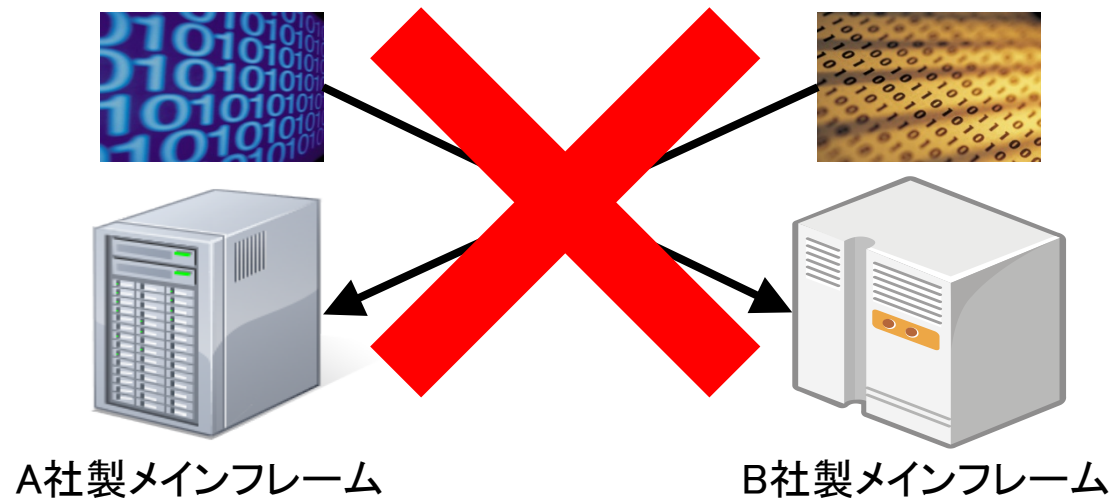
IBM 704 メインフレーム

Wikipediaより転載

- ▶ 企業の基幹業務システムとして利用される
 - ▶ 勘定系システム
 - ▶ 生産管理システム
 - ▶ 販売管理システム
 - ▶ 在庫管理システム

オープンソースソフトウェア開発の歴史 1950～1970年代（2）

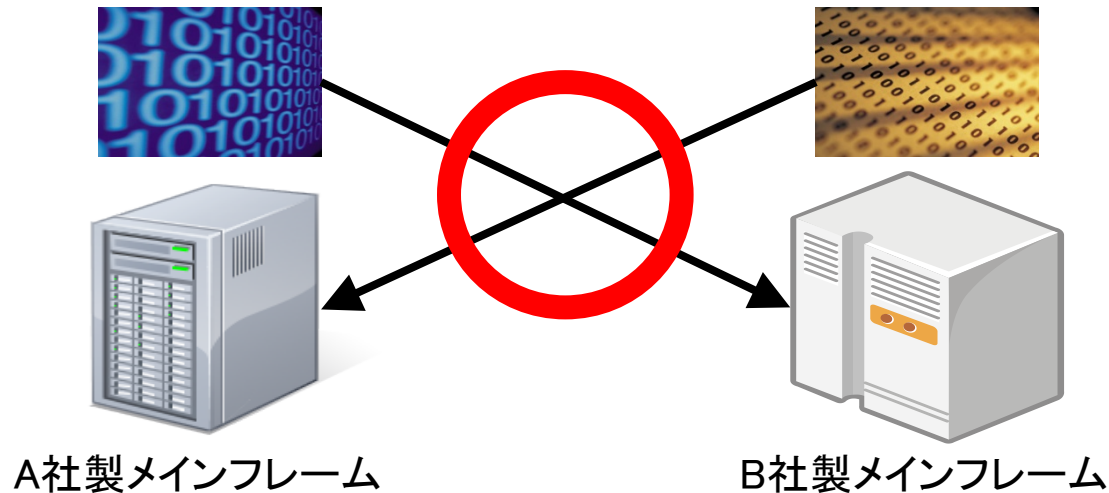
- ▶ 各メーカー独自設計のハードウェア+ソフトウェア



- ▶ ソフトウェアの互換性がほとんどなかった
- ▶ 開発費全体に占めるソフトウェアのコストは小さかった
 - ▶ ソフトウェアはハードウェアの「おまけ」
 - ▶ ソースコードが無料で付いてくるのが当たり前

オープンソースソフトウェア開発の歴史 ～1980年代（1）

- ▶ IBM System/360系がメインフレームの主流となる

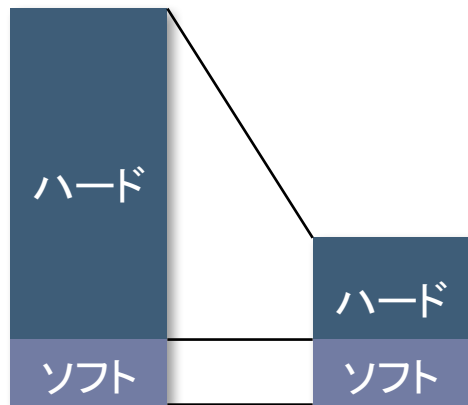


- ▶ IBM互換メーカーの台頭を招く⇒競争の激化
- ▶ 高級言語（C言語）とコンパイラの普及

他社のハードウェアでソフトウェアが動作するようになる

オープンソースソフトウェア開発の歴史 ～1980年代（2）

- ▶ 競争の激化と互換性向上の影響
 - ▶ ハードウェア価格の低下により，開発費全体に占めるソフトウェアのコストが無視できなくなる



ソフトウェアの経済価値の向上
(無料という訳にはいかなくなる)

- ▶ 技術の流出が懸念されるようになる

ソフトウェアの知的財産価値の向上
(ソースコードを付けることが難しくなる)

オープンソースソフトウェア開発の歴史

フリーソフトウェア運動：夜明け

- ▶ 現在の商用ソフトウェアの考え方が広く普及する
 - ▶ ソフトウェアは有償である
 - ▶ ソフトウェアは知的財産として保護される（ソースコードを秘匿するのは当然である）

このような時代背景を受け
一人の天才ハッカーが立ち上がる
(ブチ切れる)



オープンソースソフトウェア開発の歴史

フリーソフトウェア運動：Richard Stallman



- ▶ リチャード・ストールマン
 - ▶ 誕生：1953年3月16日（63歳）
 - ▶ 国籍：米国
 - ▶ 元・MIT AI研所属のプログラマ
 - ▶ GNUプロジェクト創始者
 - ▶ フリーソフトウェア財団（FSF）創設者
 - ▶ Emacs（テキストエディタ），GCC（Cコンパイラ）をはじめとして数々の世界的有名ソフトウェアの共同開発者

Wikipediaより転載

オープンソースソフトウェア開発の歴史

フリーソフトウェア運動：Richard Stallman

▶ なぜ切れたのか？

▶ 当時の教育研究機関

- ▶ ハードウェアに付属するソフトウェア（ソースコード）を、仲間と協力して改良・修正するのが当たり前の事だった
- ▶ ソースコードを「自由」に改良・修正する文化（いわば、オープンソース文化）
- ▶ プログラマとしてその自由な文化をこよなく愛していた

▶ 「商用ソフトウェア文化」の浸透

- ▶ 教育研究機関の「オープンソース文化」が破壊される
- ▶ ハードウェアメーカーがソースコードを秘匿するようになったため

オープンソースソフトウェア開発の歴史

フリーソフトウェア運動：GNUプロジェクト



- ▶ 1983年にリチャード・ストールマンがMIT AI研在籍中に発表したフリーソフトウェア開発プロジェクト
- ▶ GNU (Gnu is not Unix)
- ▶ 目標：完全に**フリー**なソフトウェアで構成される (Unix互換の) オペレーティングシステムを実現すること
 - ▶ **フリー (自由)**：英語のfreeは「無料」という意味合いが含まれるが、GNUプロジェクトの「フリー」には日本語の「自由」に近いlibertyの意味合いが強く込められている

オープンソースソフトウェア開発の歴史 フリーソフトウェア運動：GNUソフトウェア



- ▶ Hurd：プロジェクトにより作られているオペレーティングシステム・カーネル
- ▶ **GNU/Linux**：GNUソフトウェアとLinuxカーネルの組み合わせ+ α
- ▶ GNU libc：GNU Cライブラリ
- ▶ **Bash**：シェル（OSのコマンドを実行するインタプリタ）
- ▶ Coreutils：基本的なコマンド集
 - ▶ fileutils：ファイル操作の基本Unixコマンド
 - ▶ textutils：テキスト操作の基本Unixコマンド
 - ▶ shellutils：ユーザー情報やシェルスクリプトに有用な基本コマンド
- ▶ GNUコンパイラコレクション (GCC)
 - ▶ **Cコンパイラ**
 - ▶ C++コンパイラ
 - ▶ Objective Cコンパイラ
 - ▶ Javaコンパイラ
- ▶ GDB：デバッガ
- ▶ GNU **Emacs**：多機能エディタ、コンピュータ操作環境、ソフトウェア開発環境。
- ▶ GNU **Make**：プログラムのビルド作業を自動化するツール。
- ▶ GnuPG：フリーなPGPの実装, など

商用ソフトウェアを必要とするものはGNUソフトウェアとして認められていない

オープンソースソフトウェア開発の歴史 フリーソフトウェア運動：コピーレフト



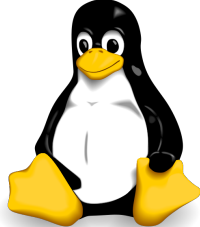
▶ GNU宣言

- ▶ 70年代までのようにソフトウェアに対するユーザの「自由」を取り戻すこと
 - ▶ ソフトウェアのソースコードを使って勉強できる自由
 - ▶ ソフトウェアを他の人々と共有できる自由
 - ▶ ソフトウェアを修正できる自由
 - ▶ 修正版を配布できる自由
- ▶ 後のOSSの著作権・ライセンスの思想に大きな影響

▶ コピーレフト

- ▶ 著作権 (copyright) とは対照的な考え方
- ▶ 著作権を放棄するのではなく、保持したまま二次的著作物も含めてすべての者が著作物を利用・再配布・改変できるようにする

オープンソースソフトウェア開発の歴史 フリーソフトウェア運動：Linuxの登場



ニュースグループへの歴史的投稿（1991年8月25日）

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Message-ID: 1991Aug25.205708.9541@klaava.Helsinki.FI
Date: 25 Aug 91 20:57:08 GMT Organization: University of Helsinki

minixを使っている皆さん、こんにちは。

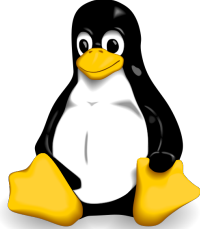
今僕は**フリーの386(486)ATクローン**で稼働するオペレーティングシステム(といっても単なる趣味で、gnuの用に巨大でプロフェッショナルなものにはならないだろうが)に取り組んでいます。4月から作り上げてきて、おおむね準備が整いました。皆さんから、**minix**の良いところ、好きでないところなどを聞きたいと思っています。僕のシステムもそれなりにminixに似ています。例えばファイルシステムの物理層などは実際的な理由でそっくりです。僕はすでに**bash(1.08)**と**gcc(1.40)**を移植し、うまく動いています。これならば、数ヶ月の間にもっと役立つツールを増やすことができるでしょう。だからこそ、**皆さんがどんな機能を欲しいと思っているのかぜひ知りたいと思います。どんな提案でも歓迎します。**ただし、実際に実装するかどうかは約束できませんが :-)

Linus (torvalds@kruuna.helsinki.fi)

PS. 誤解のないように。これはminixのコードを一切含みません。そして マルチスレッド化されたファイルシステムをもっています。これは386タスクスイッチングなどを使っているので移植性は極めて低く、AT-ハードディスク以外を今後もサポートしないでしょう。まあ、そこまでですが :-)

オープンソースソフトウェア開発の歴史

フリーソフトウェア運動：Linuxカーネル



- ▶ リーナス・トーバルズによって1991年8月に公開され、現在も開発が続くカーネル（OSの中核部分）
 - ▶ OS全体を指す場合はGNU/Linuxと呼ぶ
 - ▶ UNIX由来のソースコードを一切利用せず一から開発
 - ▶ コピーレフトのライセンスGPLに従って配布
- ▶ **バザール方式**のソフトウェア開発
 - ▶ 誰でも開発に参加できる（Linux Kernel Mailing Listへバグ報告や修正、機能拡張パッチを投稿稿できる）
 - ▶ インターネット発展期と同時期に開発を開始したこともあり、共同開発コミュニティの形成に成功した

オープンソースソフトウェア開発の歴史

フリーソフトウェア運動：Linus Torvalds



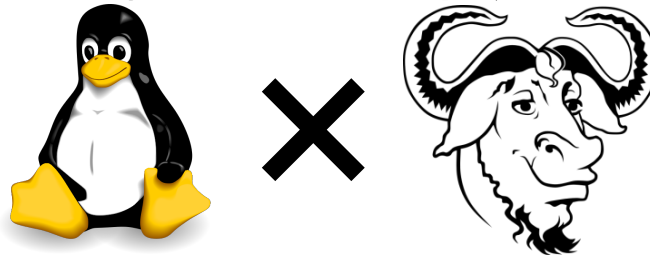
- ▶ リーナス・トーバルズ
 - ▶ 誕生：1969年12月28日（46歳）
 - ▶ 国籍：米国（フィンランド生まれ）
 - ▶ ヘルシンキ大学在学中，自作のOS（Linux）をニースグループで公開し，共同開発を呼び掛ける
 - ▶ 「慈悲深い独裁者」として，現在もLinux開発の方向性や重要事項に対する最終的な決定権を持つ

Wikipediaより転載

オープンソースソフトウェア開発の歴史 LinuxとGNUソフトウェアの融合



- ▶ LinuxとGNUの思惑の一致
 - ▶ GNUプロジェクトのOSカーネルはなかなか完成しなかった
 - ▶ LinuxプロジェクトはGNUソフトウェアの移植を行っていた
- ▶ 完全にフリーなUNIXライクOSの完成

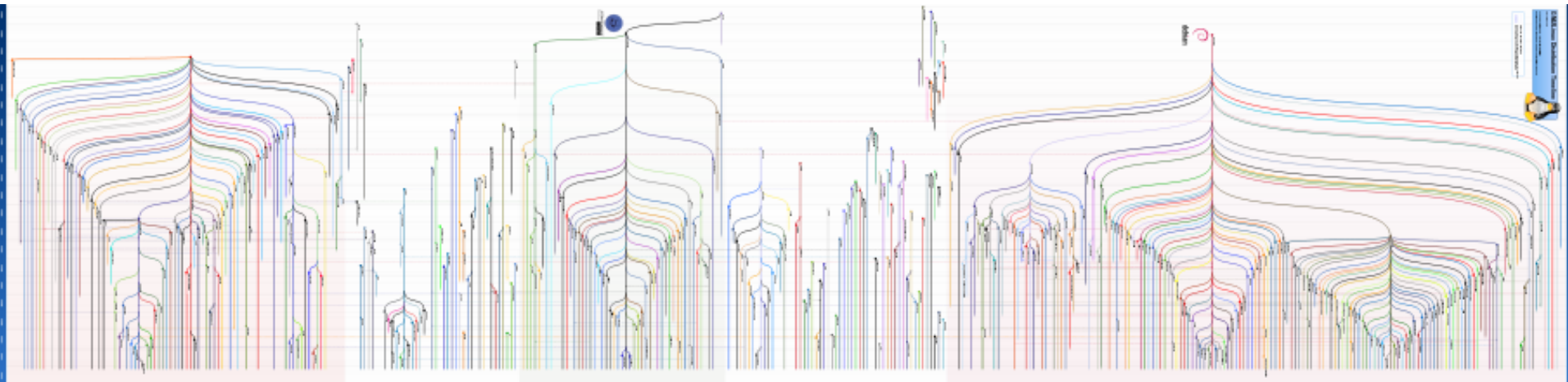


- ▶ 当時広く普及していた商用のUNIX OSを誰もが自由に利用・改良できるようになった
- ▶ GNU/Linuxをベースとした様々なLinuxディストリビューションが開発され爆発的に普及した

オープンソースソフトウェア開発の歴史

Linuxディストリビューション

- ▶ 一般ユーザが容易にインストールしたり利用できるような様々なソフトウェアを含めてパッケージとしてまとめたもの
 - ▶ FreeBSD系列のディストリビューションも存在する
 - ▶ バリエーション豊かな系譜



オープンソースソフトウェア開発の歴史

オープンソースのライセンス

- ▶ 様々なライセンス形態（と派生バージョン）
 - ▶ **GNU General Public License (GPL)**
 - ▶ GPL v1, v2, v3
 - ▶ 以下を二次的著作物にも保証するライセンス
 - プログラムの実行
 - プログラムの動作を調べ、それを改変すること
 - 複製物の再頒布
 - プログラムを改良し、改良を公衆にリリースする権利
 - ▶ 修正BSDライセンス
 - ▶ MIT License (X11 License)
 - ▶ Apache Software License
 - ▶ Common Public License (CPL)

オープンソースソフトウェア開発の歴史

オープンソースのライセンス：公開義務と競合問題

- ▶ OSSを再利用した商用SW開発における問題
 - ▶ 前提
 - ▶ SWは知財であり通常ソースコードを公開しない
 - ▶ 問題1：公開義務
 - ▶ 商用利用を目的としてOSSを再利用（改変・修正含む）した場合、OSSライセンスによっては、すべてのソースコードを公開しなければならない場合がある
 - ▶ 問題2：ライセンス競合
 - ▶ 商用利用を目的として**複数の**OSSを再利用（改変・修正含む）した場合、OSSライセンス同士がお互いにライセンス違反となる場合がある



レポート課題A：Linuxディストリビューション

▶ 課題

- ▶ 以下の3つの系列に属するLinuxディストリビューションをそれぞれ1つ以上調べ、その歴史と特徴について、それぞれA4用紙1枚程度（合計3枚）でまとめなさい

- ▶ Debian系（Ubuntuなど）
- ▶ Redhat系（CentOSなど）
- ▶ Slackware系（Slackwareなど）

▶ 加点対象

- ▶ Linuxディストリビューション4種類以上で加点
- ▶ FreeBSD系ディストリビューションも追加調査すれば加点

▶ 注意点

- ▶ 参考にした文献やWebサイトがあれば引用して明記すること！
- ▶ コピペと判断できた場合は、厳重注意の上、本レポート課題については0点とします（過去に数名います．．．）

レポート課題B：オープンソース・ライセンス

▶ 問題意識

- ▶ 複数のOSSを利用した製品開発では、異なるライセンスが適用されたOSSを組み合わせる場合があります
- ▶ ライセンス形態によっては、一方の規約が他方の規約を侵害する「ライセンス競合（破壊）」という極めて大きな問題が生じます

▶ 課題（A4用紙2~3枚程度）

- ① OSSのライセンスを1つ以上調べその特徴をまとめる
- ② 「ライセンス破壊（違反）」が生じるようなシステムの具体例（複数のOSSを組合わせたシステム）を考え、問題点と解決策を提示する
- ③ 「ライセンス破壊（違反）」が実際に生じた実例を調査して、なぜ問題となったのかの理由と、どの様に解決したのかについてまとめる

レポート課題：オープンソース・ライセンス

- ▶ レポート課題AまたはBを提出すること
 - ▶ レポート課題A・Bともに提出できれば、次回のレポート課題は免除します
- ▶ 提出先：reports@oss.sys.wakayama-u.ac.jp
- ▶ メールタイトル：ソフトウェア工学特論
- ▶ 期限：10月17日（月）12:00まで

スケジュール（大平担当分）

- ▶ 10月5日：オープンソースソフトウェア（1）
- ▶ 10月12日：休み（体育の日）
- ▶ 10月19日：オープンソースソフトウェア（2）
- ▶ 10月26日：リポジトリマイニング（1）
- ▶ 11月2日：リポジトリマイニング（2）