職業訓練教材の作成・改訂における LaTeX 及び Git の活用事例

松永 龍真　福原 卓巳　岩島 早苗

島根職業能力開発促進センター

2026-11-28

背景・目的

職業訓練のための教材の作成・改訂

職業訓練の実施にあたって

- 訓練内容に応じた教材や補助資料を作成
- 実施状況や受講者の理解度などに応じて改訂

電子情報系の離職者訓練

- 関連技術の進歩や使用するソフトウェアの更新が比較的速い
- → 教材などの継続的な見直しと改訂が必要

訓練教材の改訂履歴の管理

訓練教材の継続的な改訂にあたって
　教材の現在のバージョンだけでなく
　どのような改訂が行われたのかを把握できることが重要

ソフトウェア開発で利用される**バージョン管理の考え方**
　→ 訓練教材の作成・改訂においても適用

Microsoft Word における「変更履歴の記録」

訓練教材の作成に用いられる代表的な文書作成環境
→ **Microsoft Word**

変更履歴の記録：文書に対する加筆や修正を記録する機能
文書の校正や編集過程を確認する上では有用
文書の改訂履歴を体系的に蓄積するものではない
→ 文書のバージョンを管理するための仕組みではない

本実践の目的

職業訓練の現場における教材の作成・改訂において
　　ソフトウェア開発で用いられるバージョン管理の手法を適用

職業訓練教材の継続的な改訂・管理を実現

実際の運用において明らかとなった課題について整理する

Gitを用いたバージョン管理

バージョン管理とは

単一のファイルやファイルの集合について
　それらの編集（加筆や修正）を記録する
　必要に応じて特定の過去の状態を再現する

単純なバージョン管理の例
　変更前にファイルのコピーを作成
　変更後のファイルの名前に日付やメモを追記

ファイル名によるバージョン管理の例

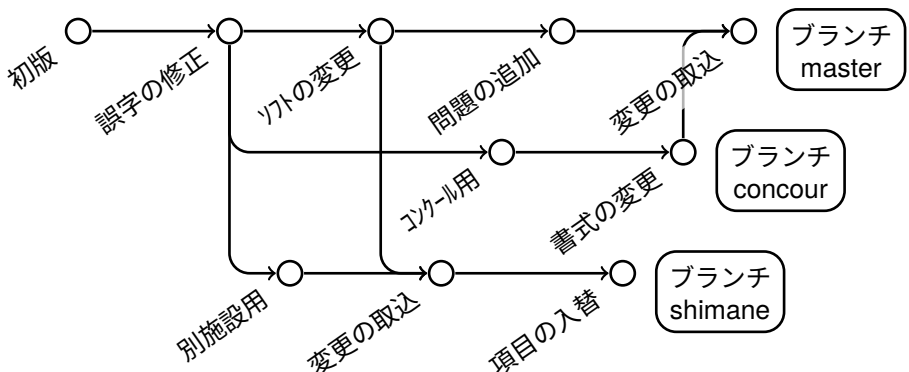
- テキスト_R08-09-08.docx
- テキスト_R08-09-08_誤字修正.docx
- メインテキスト.docx
- メインテキスト_25-02-21.docx
- メインテキスト_20250520.docx
- メインテキスト - コピー.docx
- 最新版_メインテキスト.docx

Git：ソフトウェア開発分野のバージョン管理システム

バージョン管理システム（version control system: VCS）
　変更履歴を体系的に記録・管理するシステム

Git：ソフトウェア開発の分野を中心に広く利用される VCS
　ひとまとまりの変更を**コミット**として記録
　変更者や日時、内容などを記述したメッセージを添える
過去の状態の確認や再現、バージョン間の比較を実現可能

教材を Git でバージョン管理する例



LaTeXを用いた教材作成

Git と相性の良いファイル形式

Git はテキストファイルの変更内容を差分表示することに長ける
　変更内容の比較を容易に行える形式で教材を管理
　→ Git の利点を活かせる

Microsoft Word のファイル形式（ISO/IEC 29500）はバイナリ形式
　そのままでは Git で管理するには不向き

Git と相性の良い教材の記述形式を検討

テキスト形式で文書を記述する LaTeX

LaTeX：TeX を基盤として構築されたフリーの組版システム
文書の内容及び構造をテキスト形式のソースファイルで記述
LaTeX 処理系によって組版し、PDF ファイルなどを生成

```
75 ¶
76 \section{Gitを用いたバージョン管理}¶
77 ¶
78 バージョン管理とは，¶
79 単一のファイルまたはファイルの集合に
80 必要に応じて過去の状態を再現できるよ
81 最も単純な方法として，¶
82 変更前のファイルのコピーを作成し，¶
83 ファイル名などに変更日時や変更内容に
84 保存する方法が考えられる．¶
85 しかし，この方法では，¶
86 ファイルの増加に伴って管理が煩雑とな
87 どのファイルが最新のものであるかを判
88 人為的な誤りが生じやすい．¶
89 例えば，\図{fig:最悪}のように，¶
```

2. Git を用いたバージョン管理

バージョン管理とは，単一のファイルまたは
の集合に対する変更を記録し，必要に応じて過
を再現できるようにすることである[1]．最も単
として，変更前のファイルのコピーを作成し，
名などに変更日時や変更内容に関する覚書を附
存する方法が考えられる．しかし，この方法で
イルの増加に伴って管理が煩雑となり，どのフ
最新のものであるかを判別しにくくなるなど，

文書の構造化で体裁を統一できる

LaTeX では文書の内容と体裁とを分離して記述できる

文書の構造

「ここを見出しとする」や「ここを箇条書きにする」など

文書の体裁

主に見た目に関する部分は本文と分離されたファイルに記述

複数の教材に対して統一された体裁を容易に適用できる

運用の実践

教材管理の実践

これまでの教材管理の状況

訓練用ネットワーク上のファイルサーバ
Microsoft Word 文書といくつかのコピーとを保存
→ どのファイルが最新版であるのか判りにくい
→ ファイル間にどのような差異があるのか判りにくい

LaTeX と Git とを用いた教材の作成・改訂

2026 年 2 月より約半年間の運用
訓練に用いる教材や補助資料など計 11 篇を管理

環境構築

Microsoft Windows 上で動作する環境を構築

LaTeX 処理系 TeX Live 2026

Git 処理系 Git for Windows

レポジトリの管理・共有システムを構築

仮想化環境（Proxmox VE）上に Gitea をセルフホスト

これらのシステムを訓練用ネットワーク上に構成

システム構成図



LaTeX の導入による利点と反省点

一般的な日本語文書の体裁を提供する jlreq クラスを使用
- 教材の作成者によらず統一感のある体裁を実現
- 文書の構造の整理をしやすくなった
- 目次や図番号の管理負荷を低減できた

LaTeX 記法のコマンドや LaTeX 処理系の操作方法の習得が必要
- WYSIWYG な文書作成環境に慣れた指導員にとって障壁

Gitの導入による利点と反省点

誤字脱字など些細な修正を含む詳細な変更履歴をも追跡可能に
　　コミット間の変更箇所を差分形式で確認可能

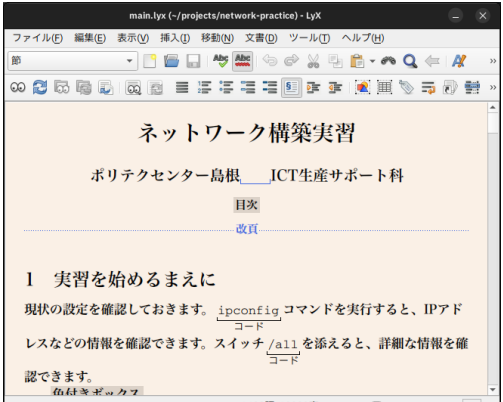
バージョンごとにコピーしていたファイルを一本化
　　大量のコピーの中から目的のものを探す手間の省力化

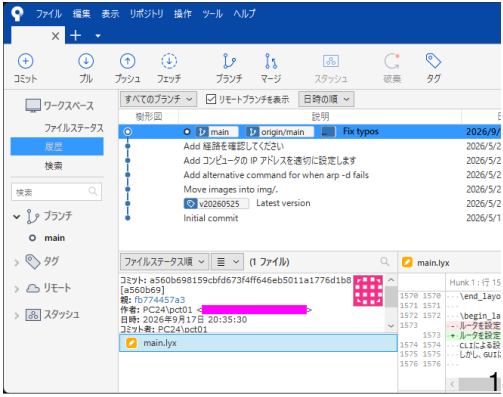
他の指導員が作成した教材に対する誤字脱字の修正や内容の充実
　　教材の保守における連携を容易に行えるように

操作のためのコマンド習得や誤操作の取消に手間取る

反省点を踏まえて

LaTeX と Git とのいずれもインタフェイスが非直感的
→ 対策としてグラフィカルツールを導入





まとめ

職業訓練教材の作成・改訂における LaTeX 及び Git の活用

職業訓練教材の作成・改訂に LaTeX 及び Git を導入する実践
- 教材をバージョン管理することは有用
- 非直感的なインタフェイスを克服するための工夫が必要

Git によるバージョン管理を科や系、施設をまたいで運用
- 教材を共有するための仕組みとしても活用可能
- → 改訂履歴を含め、組織全体で活用できる資産へ

おわり

このスライドについて

Written in November 2026.

Permanent ID of this document: `04c51c83a244dd56`.

Copyright © 2026 松永龍真, 福原卓巳, and 岩島早苗.

特記なき場合、プログラムやソースコード、その他のコンテンツは BSD 2-Clause License の下で利用可能です。なお、各画像の著作権は、それぞれの画像の著作者に帰属します。