

強力なグラフィック機能を備えた日本語組版処理システム —— Twilight: An Implementation of Graphical Typesetting ——

1. 背景

組版は、今日の情報化社会を支える上で欠かせない技術である。美しい組版の実現は、可読性や視認性の向上に大きく貢献する。特に、我々が日頃利用する日本語は、ひらがな・カタカナ・漢字・欧文といった複数の文字種や、縦組み・横組みが混在しており、その組版技術には複雑なルールが存在する。

現在、組版処理を行うソフトウェアには大きく 2 つの種類が存在する。

1 つ目は、**WYSIWYG** (What You See Is What You Get) エディタと呼ばれるソフトウェアであり、Adobe InDesign, Illustrator, Microsoft Word 等が該当する。GUI を用いて装飾性の高い文書を制作しやすい反面、一括操作や自動組版には不向きであり、文書制作における作業効率は低下する。加えて高度な日本語組版機能への対応は、一部のソフトウェアに限定されているのも現状である。

2 つ目は、マークアップ言語を用いた CLI ベースでの**組版処理システム**であり、LaTeX や SATySFi 等が該当する。文書構造をテキストベースで記述するため、文書の再利用性の向上が期待されるほか、マクロ等の作成により、高度な拡張処理・自動組版処理がサポートされる。また多くのソフトウェアで、美しい日本語組版機能がオープンソースで提供されている。一方で、主な用途は論文執筆やレポート等、テキストが主体の文書に限定されてきた。

こうした現状を踏まえると、以下の項目が課題として挙げられる。

- ・装飾性の高い印刷物（雑誌・ポスター等）の制作は、WYSIWYG に依存せざるを得ない状況が続いている。
- ・一般のユーザーは高度な組版処理を実現するソフトウェアを有しておらず、街中には粗悪な組版による文書が散見される。

2. 目的

本プロジェクトは、以下の 2 つを目的とする。

I. CLI ベースでのグラフィカルな組版処理による、効率的な組版プロセスの実現

WYSIWYG ベースでの組版処理に変わって、マークアップ言語によるグラフィカルな組版を実現し、組版処理のさらなる効率化や自動組版への対応を目指す。また、潜在的な需要を抱えているタグ付き組版の一般化を推し進める。

II. オープンソースでの美しい組版技術の提供による、組版技術発展への貢献

高度な組版機能の利用や、グラフィカルな組版を実現するには高価なソフトウェアの導入が必須であった現状を改善する。同時に、これまで組版技術に興味が薄かった層へのシステムの普及を目指し、誰もが美しい組版にアクセスできる環境を整備する。

3. 開発の内容

本プロジェクトでは、「**グラフィック機能の大幅な強化**」「**強力な日本語組版機能**」に主眼をおいた、新たな組版処理システム Twilight を開発した。Twilight は、組版言語・レンダリングエンジン（言語処理系）・周辺システム群により構築される。図 1 に本システムの構造を大まかに示した。

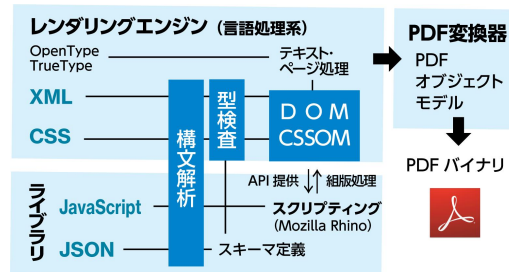


図 1：Twilight のシステム構造

3.1. 組版言語

組版言語には、XML・CSS を組版用に独自拡張した言語を利用する。XML に文書構造とテキストを記述し、CSS にはスタイル情報を定義する。

XML では、**テキストタグ**を利用した独自のテキストスタイリング機構を有する。テキストタグは入れ子構造を取る必要がないため、以下の様に XML に比べて簡潔にマークアップできる特徴を持つ。

標準 XML：XML タグによるマークアップ

```
<span class="a"> こんな </span><span class="b"> 夢 </span>
<span class="c"> を見た。 </span><span class="d"> 腕組 </span>
<span class="e"> をして枕元に </span> 坐っていると、
```

Twilight：テキストタグを用いたマークアップ

```
{a: こんな {b: 夢 :a} を見た。 {c: 腕組 :b} をして枕元に :c} 坐っていると、
```

CSS では、テキストに対する表現を中心に、標準の CSS には存在しないプロパティを実装することで、より高度なスタイリングを実現している。また、**CSS レイヤー機能**を用いることで、フィルやストロークを多層に重ねた表現が可能となり、スポーツ新聞やテロップ等にみられる複雑なスタイリングが実現される（図 2）。



図 2：CSS レイヤー機能を用いた組版例

3.2. レンダリングエンジン

レンダリングエンジンは、組版言語を処理して PDF ファイルを出力する Java プログラムである。型システムや JSON を用いたスキーマ定義機能の搭載により、効率的な文書作成やライブラリ開発を実現している。レンダリングエンジンの開発にあたっては、JavaScript インタプリタを除いたほぼすべての機能をフルスクラッチで実装した。

3.3. ライブラリ

ライブラリは、JavaScript と JSON から構成されるパッケージ群である。ユーザーは、ライブラリを通じて**拡張 XML・CSS プロパティ・テキストコマンド**として独自機能を実装可能である。これにより本システムの自在な柔軟性・拡張性が実現される。

需要が見込まれる機能に関しては、標準ライブラリとして実装を行った。例えば、日本語組版では追い込み・追い出し・ぶら下がりといった行分割処理を筆頭に、ルビ・縦中横・割注・字取りといった高度な機能にも対応することで、W3C「日本語組版処理の要件」に準拠した組版を実現した（図 3）。欧文組版では、Knuth-Plass Breaking Paragraphs into Lines Algorithm の採用により、高品質な行分割処理を実装している。

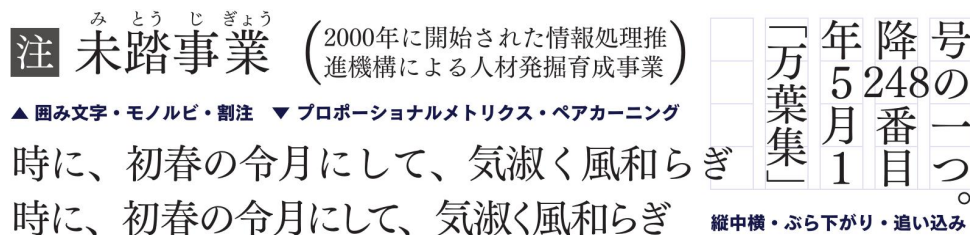


図 3. Twight で実現可能な日本語組版の例

ページレイアウトにおいても、段組み・段抜き・字取り処理や版面計算の自動化等、柔軟なレイアウトの実現をサポートしている。外部プロセスとの連携により、Twight で制作した紙面中に、LaTeX の数式組版を挿入するといった応用も可能となる（図 4）。

$$\text{二次方程式の解の公式は } x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \text{ と表される。}$$

図 4：LaTeX による数式組版を Twight で制作した文書に挿入した組版例

3.4. Web 組版システム

オンラインエディタとビューワを一体化した、Web 組版システムの開発を行った。システムはブラウザ上で作動する（図 5）。PC やタブレット端末といったマルチデバイスでの組版処理を実現する、次世代の組版フローを見据えたシステムである。

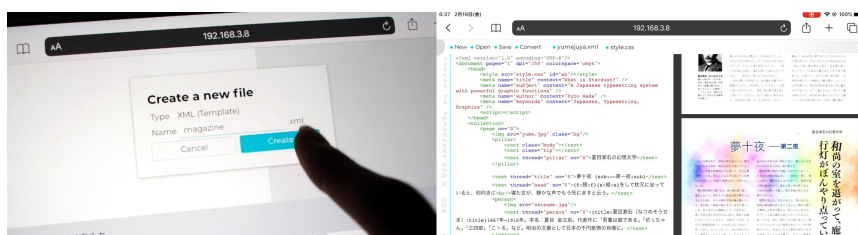


図 5：Web 組版システムの利用イメージとスクリーンショット

4. 従来の技術（または機能）との相違

従来の組版処理システムでは実現が困難であった、グラフィカルな文書の制作に対応した。図 6 に Twight を用いて実際に制作した印刷物を示した。また、開発詳細（本文書）や開発概要の組版・図版制作にも Twight を利用している。



図 6：Twight を用いて制作された組版結果

Twight の開発においては「機能は可能な限りライブラリで実装する」といった方針を打ち立てている。言語自体での機能実装は最小限に留めることで、印刷物に要求される互換性の確保を実現した上で、ユーザーが介入可能な処理を最大限まで拡大した。

5. 期待される効果

Twight では、テキストベースでの入力、グラフィカルかつ美しい組版を備えた媒体制作を実現するため、従来の組版処理システムの対象から外れていた、雑誌・ポスター・リーフレット・小冊子等の組版も可能となる。加えて、動画編集におけるテロップ作成、サインシステムの整備、LCD への埋め込み等、印刷物制作を越えた幅広い用途が想定される。

その他、Twight では幅広い API をユーザーに開放している。この特徴により、組版言語あるいは標準ライブラリではカバーしきれない細やかなニーズやハウスルールに対して、ユーザー自身が対応可能となる。有志によるライブラリ開発も期待でき、本システムの持続的な発展が実現される。

6. 普及（または活用）の見通し

Twight はオープンソースでの公開を予定しており、既存組版処理システム（LaTeX 等）の利用者や Web 開発者をはじめとする幅広いユーザーへの普及を目指す。また今後、より柔軟な自動組版への対応や多言語組版のサポートも視野に、ライブラリの拡充を進めていく所存である。

7. クリエータ名（所属）

和田 優斗（横浜市立横浜サイエンスフロンティア高等学校）