

# シルエット Web ブラウザにおける 多重透明レイヤー管理機構とその応用

杉山 亮弘<sup>1,a)</sup> 丹羽 佑輔<sup>1</sup> 白松 俊<sup>1</sup> 大園 忠親<sup>1</sup> 新谷 虎松<sup>1</sup>

概要：既存のアプリケーションの多くは、特定の作業に特化したものが多い。ユーザはアプリケーションを利用するとき、Web 上で使い方を学ぶなど、複数のアプリケーションを同時に使うことが多い。それに伴い、他のアプリケーションの情報と連携して、ユーザの作業を支援しようとする研究が多く行われている。本研究では、多重透明レイヤーをデスクトップ上に重畳表示することが可能な WebKit ベースの Web ブラウザであるシルエット Web ブラウザを提案し、その応用例を述べる。シルエット Web ブラウザの特筆すべき点は、マウスイベントおよび背景が透過可能なコンテンツを表示可能な点である。さらに、コンテンツを複数ユーザ間で共有することが可能である。シルエット Web ブラウザにより、新たな協調作業支援が可能になる。シルエット Web ブラウザの応用例として、デスクトップ PC とタブレット端末を利用した添削システムを示す。

キーワード：シルエット Web ブラウザ、レイヤーモデル、多重透明レイヤー

## 1. はじめに

既存のアプリケーションの多くは、特定の作業を行うことに特化したものが多い。例えば、Microsoft 社の Word は、文書処理に特化している。ユーザは、アプリケーションを使うとき、ウェブ上で入門ページを閲覧、エラーの対処、アプリケーションの使い方などを調べる。ユーザは作業を行うとき、複数のソフトウェアを利用することが多い。それに伴い、他のアプリケーションの情報と連携して、ユーザの作業を支援しようとする研究が行われている [4]。

本研究では、後述する背景およびマウスイベントが透明な Web ブラウザを利用して既存のアプリケーションと連携して研究活動を支援する環境を提案する。既存のアプリケーションと連携して研究支援をするとき、ユーザへの干渉の度合いを制御することが重要である。

本研究では、既存のデスクトップのレイヤー構造にウィンドウレベルの異なるレイヤーを追加する。レイヤーを追加することで、ユーザは、既存の表示領域に加えて、デスクトップの背景、最前面、の領域に情報を表示することが可能になる。このレイヤーは、複数の機能を持つ情報表示機構を持つ。この機構を利用することで、レイヤー上に Web

コンテンツを表示することが可能になる。

本稿では、2 節で、マウスイベントが透明な Web ブラウザであるシルエット Web ブラウザについて述べる。3 節では、シルエット Web ブラウザで作成されるレイヤー上を持つフレームについて述べる。4 節では、シルエット Web ブラウザを応用したシステムについて述べる。

## 2. 多重透明レイヤーを利用したシルエット Web ブラウザ

シルエット Web ブラウザは、多重透明レイヤーをデスクトップ上に重畳表示することが可能な WebKit ベースの Web ブラウザである。シルエット Web ブラウザの特筆すべき点は、マウスイベントおよび背景が透過可能なコンテンツを表示可能な点である。すなわち、操作できない半透明のコンテンツをデスクトップ上に表示することが可能である。さらに、コンテンツを複数ユーザ間で共有することが可能である。シルエット Web ブラウザにより、新たな協調作業支援が可能になる。本節では、シルエット Web ブラウザにおける多重透明レイヤー（以降、単にレイヤーと記す）について説明する。

図 1 にシルエット Web ブラウザの実行例を示す。図の A は、既存の PDF ビューアであり、B,C,D はシルエット Web ブラウザが持つフレームである。B は後述するファイル共有することが可能なフレームである。C は、PDF

<sup>1</sup> 名古屋工業大学大学院情報工学専攻  
Department of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology  
a) sakihiro@toralab.org



図 1 シルエット Web ブラウザの実行情例

Fig. 1 An example of execution of Silhouette Web Browser .

ビューアにフレームを重ね、フレーム内の情報に対して、別のユーザからのフィードバックが表示されている。D は、フレーム上に既存の Web ページを表示している。シルエット Web ブラウザのフレームは、既存のアプリケーションの前面、背面を選択して表示することが可能である。既存のアプリケーション上と併用することで、協調作業に利用することが可能である。

### 2.1 レイヤーとフレーム

シルエット Web ブラウザにおいて、計算機のデスクトップ画面（以降、デスクトップと記す）上に表示されたアプリケーションのウィンドウの前後に、シルエット Web ブラウザによるコンテンツ表示のための透明なレイヤーを導入する。

レイヤーは、その領域内にフレームと呼ばれる任意個の部分領域を持つ。ユーザは、これらのフレームにコンテンツや Web アプリケーションを表示させる。フレームの領域の形状は、矩形のみならず、多様な形状が可能である。例えば、星形や雲形のフレームによりユーザへの訴求力を制御することができる。フレーム間は、レイヤー内での前後関係を持つ。また、複数のフレームを、一体化してグループ化することで、あるフレームに追従して移動するようなフレームも実現可能である。さらに、フレーム間における仮想的な物理法則（例えば引力など）を割り当てることで、フレーム間における関係に応じた動作を実現可能である。

シルエット Web ブラウザにおいて特筆すべき点は、ユーザへの干渉の度合いを制御するために、フレームは、イベント透過可否、背景透明度、およびレイヤーを調整することが可能な点である。イベント透過可否の制御により、フレームはマウスによって操作可能か否かを制御可能である。例えば、マウスイベントを透過するフレームを再背面に表示させることで、デスクトップ画像に一体化したかのようなコンテンツ表示が可能になる。背景透明度の制御により、他のフレームやアプリケーションに対して、コンテンツを重畳表示することが可能になる。例えば、他のフレ

表 1 レイヤーの属性

Table 1 The attributes of Layers .

	イベント透過設定	ウィンドウレベル
$L_P$	TRUE	kCGScreenSaverWindowLevelKey
$L_A$	TRUE	kCGNormalWindowLevelKey
$L_W$	TRUE	kCGBackstopMenuLevelKey
$L_F$	TRUE	kCGDesktopIconWindowLevelKey
$L_D$	FALSE	kCGDesktopWindowLevelKey

ムやアプリケーションに対して補足的な情報表示をすることに利用可能である。またフレームは、動的にレイヤーを変えることが可能である。例えば、状況によって、デスクトップ画面の最前面にフレームを表示させたり、再背面にフレームを表示させたりすることが可能である。

フレームのイベント透過可否は、フレームのイベントが透過可否か、とレイヤーのイベントが透過可否かで決まる。レイヤーの属性を表 1 に示す。レイヤーは、基本的にイベントを透過可能である。 $L_D$  のみ、マウスイベントを透過する。ウィンドウレベルは、レイヤーがデスクトップのどの深さにあるかを示している。フレームのイベントの透過の可否を  $x_F$ 、レイヤーのイベントの透過の可否を  $x_L$  とすると、あるフレームの透過の可否  $x$  は、次式で表される。

$$x = x_F \wedge x_L$$

フレームは、ユーザ間で共有可能である。シルエット Web ブラウザを用いた、協調作業支援環境が実現可能である。例えば、複数ユーザ間における添削支援において、共有されたフレーム内に論文の PDF を表示し、そのフレーム上に手書き等のメモ機能を持つフレームを一体化させることで、容易に添削支援環境が実現できる。また、手書き機能を持つフレームをデスクトップ PC とタブレット端末で共有することで、タブレット端末のタッチパネルを有効利用した手書きメモが、追加のプログラミング不要で可能になる。

### 2.2 レイヤーモデル

本節では、シルエット Web ブラウザにおける多重透明レイヤーの構造について説明する。すなわち、既存のデスクトップのレイヤー構造にシルエット Web ブラウザによる新たなレイヤーを追加した、新たなレイヤーモデルを提案する。

シルエット Web ブラウザにおけるコンテンツ表示に関して、ユーザへの干渉の度合いを制御可能な点が、特筆すべき点である。例えば、ユーザに対する最小限の干渉によりユーザの作業を妨害しないコンテンツ表示から、ユーザに対して積極的に訴求するように干渉するコンテンツ表示も可能である。本システムでは、5 層のレイヤーを導入することで、本目的を実現する。すなわち、フレームに表示する情報の内容等に応じて、フレームを表示するレイヤー

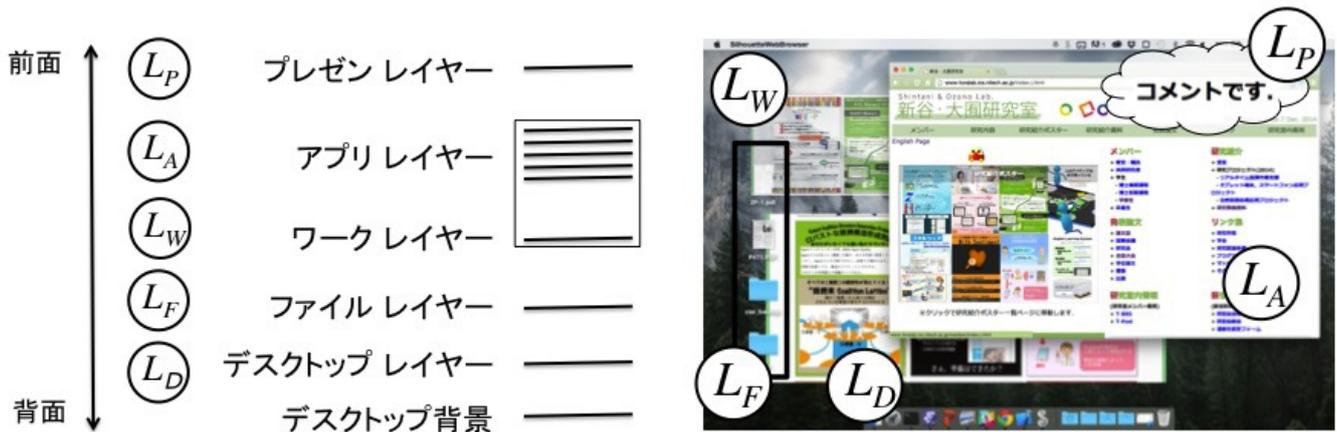


図 2 レイヤーモデルの概念図  
Fig. 2 Overview of Layer Model .

を変化させることで、ユーザへの訴求方法を制御し、ユーザへの干渉の度合いを調整することが可能になる。

シルエット Web ブラウザにおけるレイヤーは、前面から背面に向かって、プレゼンレイヤー (Presentation Layer,  $L_P$ ), アプリレイヤー (Application Layer,  $L_A$ ), ワークレイヤー (Work Layer,  $L_W$ ), ファイルレイヤー (File Layer,  $L_F$ ), およびデスクトップレイヤー (Desktop Layer,  $L_D$ ) の 5 層から構成される。図 2 は、本研究におけるレイヤーの概念図である。ここで、 $L_P$ ,  $L_W$ , および  $L_D$  の 3 層が、シルエット Web ブラウザにおいて新たに追加されたレイヤーである。既存のレイヤー構造に、シルエット Web ブラウザによる新たなレイヤーを追加することで、フレームに表示する Web ページの内容や用途に応じて、ユーザがフレームを作成/表示するレイヤーを選択することが可能になる。

図 2 の左図は、デスクトップ上に存在するレイヤーの前後関係を示した図である。左図の左の矢印は、レイヤーのウィンドウレベルを示している。前面に近いレイヤーほど、デスクトップ上では、前面に表示される。背面に近いレイヤーほど、デスクトップ上では、背面に表示される。左図の場合、 $L_P$  がデスクトップ上の最前面に表示され、デスクトップ背景が最背面に表示される。

図 2 の右図は、シルエット Web ブラウザと既存のアプリケーションを同時に利用した時のデスクトップの図である。複数のウィンドウが重なりあって表示されている。右図の記号 ( $L_P$ ,  $L_A$ ,  $L_W$ ,  $L_F$ ,  $L_D$ ) は左図の記号のレイヤー上にウィンドウまたはフレームがあることを示している。右図のデスクトップ上にあるウィンドウまたはレイヤーの前後関係は、左図のレイヤーの順序に従う。右図の  $L_P$  は  $L_P$  上のフレームを示しており、右図の  $L_W$  は  $L_A$  上のウィンドウを示している。

フレームを用いてユーザのデスクトップ上に Web ページを表示する時、(1) ユーザの作業を妨害しない、(2)

ユーザが表示した Web ページに気付く、という 2 点が重要となる。レイヤーを 5 つの異なるウィンドウレベルに作成することで、フレームに表示する情報の内容に応じて、フレームを表示するレイヤーを任意のレイヤーに変えることが可能になる。

### 2.3 プレゼンレイヤー ( $L_P$ )

$L_P$  は、デスクトップの最前面に表示されるレイヤーである。 $L_P$  は、図 2 の右図の  $L_P$  のように、デスクトップ上にあるウィンドウの中で最前面に表示される。そのため、ユーザが計算機を操作している時に、 $L_P$  上のフレームに表示されたコンテンツは、必ずユーザの目に付くと考えられる。

$L_P$  は、マウスイベントを透過可能である。 $L_P$  に作成/表示されるフレームは、デスクトップの最前面に表示される。 $L_P$  の持つフレームは、ユーザが計算機を操作している場合、必ずユーザの目につく場所に表示される。 $L_P$  上に表示するフレームが増加すると、 $L_P$  よりも背面のレイヤーの表示領域が狭くなる。特に、ユーザは  $L_A$  で作業を行うことが多いため、 $L_A$  の表示領域が狭くなると、ユーザの作業の妨害となる。そのため、 $L_P$  は、ユーザへの通知などのユーザに確実に情報を表示することが望ましい。マウスイベントを透過することで、 $L_P$  上のコンテンツを閲覧しながら、 $L_A$  上のアプリケーションを操作することが可能である。

### 2.4 ワークレイヤー ( $L_W$ )

$L_W$  は、アプリケーションのウィンドウが表示される  $L_A$  の最背面に表示されるレイヤーである。 $L_W$  は、図 2 の右図の  $L_W$  のように、ユーザが開いているアプリケーションのウィンドウの中で、最背面に表示される。 $L_W$  は、既存のアプリケーションのウィンドウの中で最背面に表示されるため、ユーザの作業を妨害することなくコンテンツを表

示することが可能である。

$L_W$  は、マウスイベントを透過可能である。 $L_W$  に作成/表示されるフレームは、 $L_A$  の最背面に表示される。フレームは、後述するフレーム機能の中で、既存のアプリケーションと連携することが可能である。そこで  $L_W$  上に、フレームを作成し、後述するデスクトップのスクリーンショットを取得するフレームの機能を利用することで、特定のアプリケーションの画面共有を行うことが可能になる。

## 2.5 デスクトップレイヤー ( $L_D$ )

$L_D$  は、デスクトップ背景の前面に表示されるレイヤーである。 $L_D$  は、図 2 の右図の  $L_D$  のように、デスクトップの最背面に表示される。 $L_D$  は、デスクトップの最背面に表示されるため、ユーザの作業を妨害することなくコンテンツを表示することが可能である。 $L_D$  は、壁紙に近いウィンドウレベルが設定されており、デスクトップを表示するショートカットキーを用いることで、 $L_D$  上のコンテンツを閲覧することが可能である。

$L_D$  は、マウスイベントを透過する。後述する任意のユーザと情報を共有するフレームは  $L_D$  に作成される。デスクトップの背景を表示するショートカットキーを利用することで、 $L_D$  上に表示されたコンテンツを閲覧することが可能である。 $L_D$  に作成/表示されるフレームは、 $L_A$  の背面に表示される。ユーザは、 $L_A$  上で作業を行うことが多い。そのため、 $L_D$  は、 $L_W$  同様  $L_A$  よりも背面にあるため、ユーザの作業を妨害しない。 $L_D$  は、 $L_A$  上にウィンドウが多い場合、表示領域が狭くなる。そのため、 $L_D$  は、カレンダーなどの緊急性の低い情報を表示することが望ましい。

## 3. フレーム

本節では、フレームについて述べる。フレームは、シルエット Web ブラウザによるコンテンツ表示するための透明なレイヤー上に、Web コンテンツや Web アプリケーションを表示可能な領域である。フレームは、シルエット Web ブラウザを起動しているユーザ間で共有することが可能である。フレームは、コンテンツを表示するだけでなく、フレーム間でのインタラクション、デスクトップとの連携が可能である。フレームの機能を組み合わせることで、新たな機能を持つフレームとして利用することが可能になる。

### 3.1 フレームとは

フレームは、レイヤー上に作成される任意個の部分領域である。フレームの領域の形状は、矩形のみならず、多様な形状が可能である。フレーム間は、レイヤー内での前後関係を持つ。フレームは共有するユーザの範囲をペア共有フレーム、パブリック共有フレーム、プライベート共有フレームの 3 つの属性で設定することが可能である。フレームは、追跡フレーム (RF)、タイムマシンフレーム (TMF)、

スクリーンショットフレーム (SSF) とファイル共有フレーム (FSF) の要素機能を持つ。

RF は、ユーザが起動している既存のアプリケーションのウィンドウを追跡する。TMF は、あるフレームが過去にどのようなコンテンツを表示していたかを表示することが可能なフレームである。SSF は、フレームがあるデスクトップ上の領域のスクリーンショットを取得するフレームである。FSF は、シルエット Web ブラウザを起動しているユーザ間でデスクトップ上でファイル共有を行うことが可能なフレームである。

フレーム毎に、ユーザが利用したい要素機能を利用することが可能である。フレームは、シルエット Web ブラウザによって管理されているため、フレーム間で情報のやりとりを行うことが可能である。フレームの要素機能を組み合わせることで、複雑な機能を実現することが可能である。例えば、SSF とファイル共有フレームを組み合わせることで、シルエット Web ブラウザを起動しているユーザの計算機を、共有の計算機のように利用することが可能である。

### 3.2 フレームの共有

フレームは、共有するユーザの範囲の属性で設定することが可能である。共有する範囲は、1 対 1、3 人以上、不特定多数に分類され、それぞれの属性を、ペア共有フレーム、プライベート共有フレーム、パブリック共有フレームと呼ぶ。

ペア共有フレームは、すべてのレイヤー上に作成することが可能である。ペア共有フレームは、あるユーザがフレームを共有したユーザに招待を送ることでユーザとフレームを共有する。

プライベート共有フレームは、すべてのレイヤー上に作成することが可能である。プライベート共有フレームは、フレームにトークンと名前を設定する。設定したトークンと名前を知っているユーザ間でフレームの共有する。例えば、グループ内のスケジュールを  $L_D$  上のプライベート共有フレームに表示することが可能になる。

パブリック共有フレームは、シルエット Web ブラウザを利用している全てのユーザ間でコンテンツを表示する。パブリック共有フレームは  $L_D$  上に作成することが可能である。パブリック共有フレームは、表示非表示を切り替えることができる。ユーザは全てのユーザとファイルを共有したくない場合、非表示にすることで、ファイル共有をやめることが可能である。

### 3.3 フレーム間のインタラクション

シルエット Web ブラウザによるインタラクションの図を図 3 に示す。シルエット Web ブラウザは、URL スキームを用いてフレームとのインタラクションを行う。シルエット Web ブラウザはフレームに指定された Web コンテンツ

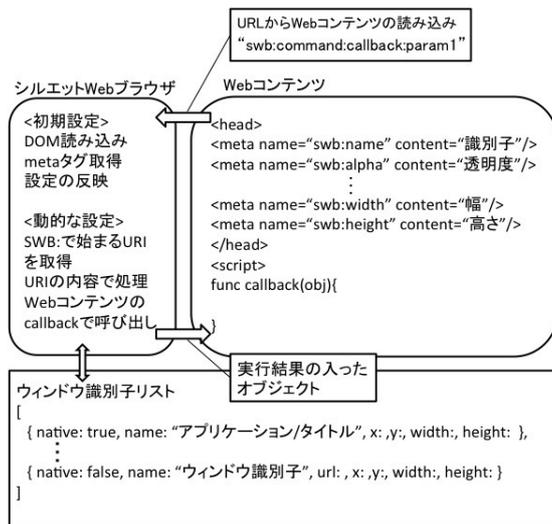


図 3 シルエット Web ブラウザによるインタラクション

Fig. 3 An example of interactions with Silhouette Web Browser .

を読み込み、Web コンテンツの HTML 内の meta タグを探し、meta タグの設定をシルエット Web ブラウザに反映する。meta タグの要素は、name, alpha, x, y, width, height を持ち、それぞれ、ウィンドウの識別子、透明度、左上部の x 座標、左上部の y 座標、幅、高さに対応する。ウィンドウ識別子リストは計算機が起動しているネイティブアプリケーションとシルエット Web ブラウザの持つ透明レイヤーの識別子のリストである。ネイティブアプリケーションを開いている場合、native 要素は true となる。

シルエット Web ブラウザの設定は、動的に変更可能である。"swb:command:callback:引数 1:...:引数 n"で指定した URL を読み込み、シルエット Web ブラウザの設定を変更する。command は、[name, alpha, x, y, width, height, capture, windows] がある。それぞれ、識別子、透明度、x 座標、y 座標、幅、および高さの設定 Web コンテンツの画像、およびウィンドウリストの取得を行う。

シルエット Web ブラウザ内のフレーム間もインタラクションが可能である。"swb:send:callback:宛先ウィンドウ識別子:送信データ"の形式で URL を読み込むことで宛先ウィンドウ識別子を持つフレームにデータを送信することが可能である。シルエット Web ブラウザは、指定されたフレーム上で送信データを引数とする callback を実行する。

### 3.4 スクリーンショットフレーム (SSF)

SSF は、シルエット Web ブラウザを起動しているユーザと画面共有を行うことが可能である。SSF は、シルエット Web ブラウザを起動しているユーザのデスクトップのスクリーンショットを取得する。取得したスクリーンショットはサーバに送信される。画面共有を行いたいユーザは、サーバ上に送信されたスクリーンショットを閲覧することで画面共有を行うことが可能である。



図 4 SSF による画面共有をする範囲の選択

Fig. 4 A range selection of screen sharing with Screen Shot Frame .

SSF は、シルエット Web ブラウザを起動しているユーザのデスクトップのスクリーンショットを取得する。スクリーンショットを取得する範囲はSSFが表示されているデスクトップ上の領域である。ユーザは、SSFを、画面共有したい領域と同じ大きさ、同じ座標にすることで、画面の一部を画面共有することが可能である。

取得したスクリーンショットはサーバに送信される。シルエット Web ブラウザを起動しているユーザと画面共有を行いたいユーザは、Web上でサーバにアクセスしサーバに送信されたスクリーンショットを閲覧する。Web上で閲覧することが可能なため、Webブラウザを持つ端末であれば、ファイル共有を行うことが可能である。

SSFを用いて画面共有を行う範囲を選択した図を図4に示す。図中Aが、ユーザが作成したSSFである。SSFをデスクトップ上に表示されている文書の一部を選択している。SSFで選択した部分は、サーバ上に送られる。図中Bでは、別の端末からWebサーバにアクセスし、画面共有を行っている。

### 3.5 ファイル共有フレーム

FSF は、シルエット Web ブラウザを起動しているユーザ間でファイル共有を行うことが可能である。パブリックフレームにファイル共有機能を持たせた場合、シルエット Web ブラウザを起動している全てのユーザとファイル共有を行うことが可能である。プライベートフレームにファイル共有機能を持たせた場合、プライベートフレームを共有しているユーザ間でファイル共有を行うことが可能である。ファイルの共有方法は、資料の配付、ユーザ間の資料の統一、資料の上書きがある。

資料の配付は、あるユーザのデスクトップ上にあるファイルをFSFを用いて別のユーザのデスクトップ上に送信する。送信先の座標は、ファイル送信者のデスクトップ上の座標と同じ座標に送信される。

ユーザ間の資料の統一は、シルエット Web ブラウザが起動しているユーザ間で同様のファイルを共有する。シルエット Web ブラウザを起動している二人のユーザが異なるファイルを共有する場合、ユーザ間の資料の統一は、

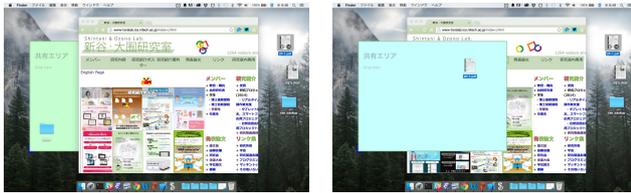


図 5 FSF の設定 (左図) .

ドラッグ&ドロップでファイル共有 (右図)

Fig. 5 The configuration of File Sharing Frame(left) .  
 An example of file sharing with drag-and-drop(right) .

ファイルの和集合を二人のユーザ間で共有する．資料の上書きは，最後にファイル共有を行ったユーザのファイルを既存のファイルに上書きする．既存の共有しているファイルと，新しく共有するファイルが異なる場合，新しく共有するファイルをユーザ間で共有する．FSF を使ったユーザ間の資料の統一の実行例を図 5 に示す．図 5 の左図は，FSF をデスクトップ上に作成した図である． $L_D$  に表示されている領域が FSF である．ここでの，FSF は  $L_D$  に含まれるためマウスイベントを取得しない．

図 5 の右図は共有エリアを利用している図である．左図では  $L_D$  に表示されていた FSF が，ファイルをドラッグすることで  $L_W$  に移動する．FSF にファイルをドロップすることで，シルエット Web ブラウザを起動している他のユーザとファイルを共有することが可能である．

## 4. 応用

シルエット Web ブラウザの応用例として，アプリケーションなどの既存のシステムとシルエット Web ブラウザが連携するシステムを示す．ここでは，シルエット Web ブラウザの応用例として添削システムを示す．添削システムでは，ユーザは SSF と FSF を用いて添削範囲と添削元のファイルを共有する．

### 4.1 添削システム

論文，発表資料などの，研究資料の作成において研究資料の添削をすることは，研究資料の作成の基本的な作業である [2]．紙の資料を共有して添削を行う場合，言葉で詳細な添削，ジェスチャーを通じた添削，手書きでの添削を行うことができる．計算機上で添削を行う場合，マウスとキーボードを用いて添削を行うため，紙ベースの添削に比べ添削の表現の幅が狭くなる．タブレット端末を用いて，電子書類にアノテーションを付与する研究 [3] も行われている．タブレット端末で，添削を行う場合，画面サイズの制約上，書類の全体を把握することが困難である．

本研究では，シルエット Web ブラウザを利用して，デスクトップ PC とタブレット端末を用いて添削を行うシステムを提案する．シルエット Web ブラウザは Web コンテンツをデスクトップ上にオーバーレイするブラウザであ

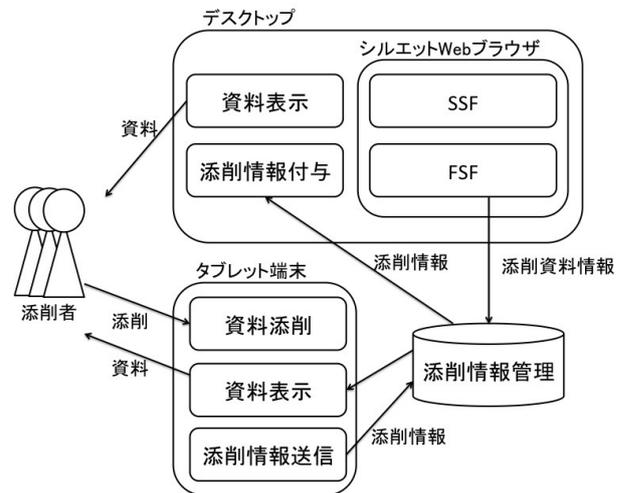


図 6 添削システムのシステム構成図

Fig. 6 The system configuration diagram of Editing System .

る．シルエット Web ブラウザを用いた添削システムを実現することで，表示する Web コンテンツは端末に依存することなく表示することが可能である．デスクトップ PC とタブレット端末の二つの計算機を用いることで，デスクトップ上で資料の全体を見渡し，タブレット端末で添削したい部分に対して細かく添削を行うことが可能となる．タブレット端末を用いることで，タッチ入力を行うことが可能となる．

シルエット Web ブラウザを用いた添削支援システムの構成図を図 6 に示す．本システムは，デスクトップ PC とタブレット端末の二つの計算機を用いる．デスクトップ PC 上のシステムは，資料表示，添削情報付与の 2 つの機構とシルエット Web ブラウザ内の SSF, FSF の 2 つの要素機能から構成されている．タブレット端末上のシステムは，資料添削，資料表示，添削情報送信の 3 つの機構から構成されている．ユーザに添削対象の資料を表示する．ユーザは表示された資料から添削したい部分をシルエット Web ブラウザを用いて選択する．シルエット Web ブラウザによって選択された資料の一部をタブレット端末で添削する．添削者が添削した情報は，デスクトップ PC 上の資料に反映される．

デスクトップ PC 上の資料表示機構では，添削して欲しい研究資料をユーザに表示する．添削者によって，研究資料が添削されている場合，添削情報付与機構から添削情報を受け取る．受け取った添削情報を元の研究資料に反映し，添削情報付きの研究資料をユーザに表示する．添削情報付与機構では，添削者がタブレット端末を用いて添削した情報をサーバから受け取る．受け取った情報を資料表示機構に渡す．

シルエット Web ブラウザを拡張し，添削範囲選択機構と添削範囲送信機構を新たに作成した．添削範囲選択機構では，添削者は，シルエット Web ブラウザのレイヤーを



図 7 研究資料の添削例

Fig. 7 An example of editing a research document .

用いて、添削範囲を選択する。添削者が指定した添削範囲の情報は、画像化され添削範囲送信機構に渡される。シルエット Web ブラウザはやりとり可能でサイズが可変なレイヤーを複数持っている。添削者は、シルエット Web ブラウザのレイヤーを使って添削範囲を選択する。シルエット Web ブラウザは、添削範囲の画面上のスクリーンショットを取得する。添削範囲送信機構では、添削範囲選択機構で添削者が選択した添削範囲の画像をサーバに送信する。

タブレット端末上の、資料表示機構では、添削者がデスクトップ PC 上でシルエット Web ブラウザを用いて取得した選択範囲を添削者に表示する。添削者は、資料表示機構で表示された研究資料に対して添削を行う。資料添削機構では、添削者が研究資料に添削した内容をサーバに送信する。

研究資料の添削例を図 7 に示す。左図はデスクトップ PC 上の図である。右図は、タブレット端末上の図である。添削者は、タブレット端末上でシルエット Web ブラウザを用いて選択した部分の添削を行う。添削者は、タブレット端末下部のボタンから、添削に用いるツールを選択することができる。添削に使えるツールは、フリーハンド、テキスト、図形、色の選択がある。

フリーハンドで添削を行うことで、曲線や複雑な図形など、作成が困難なアノテーションをユーザが自由に作成することが可能となる。テキストを用いて、添削を行うことで、手書きでは、書けないような細かい文字も使うことが可能となる。図形のアノテーションは、丸、四角、直線から選択することが可能である。フリーハンドは複雑な図形を作成することが可能であるが、手での入力となるため、まっすぐな線を引くことは困難である。図形アノテーションを用いることで、丸や四角などの簡単なアノテーションを添削に利用することが可能である。添削者がタブレット端末を用いて添削した情報は、デスクトップ PC 上に送られる。デスクトップ PC 上では、表示されている研究資料上に添削者がタブレット端末から添削した情報が表示されている。

## 5. 関連研究

Fourney らは、Web ブラウザ内の情報と関連性のデスク

トップ上のアプリケーションを結びつけるシステム Inter-Twine を開発した [4]。ユーザがアプリケーションと Web ブラウザを利用して、情報を再検索することを支援するために、Web ブラウザ内の情報とアプリケーションの結びつきの履歴を作成した。

Yoon らは、電子書類に対して、対面会議などで起きるコミュニケーションが可能なシステム RichReview を開発した [3]。RichReview は、手書き、音声解説、指示的なジェスチャーの三つの方法で、電子書類上にアノテーションを作成することが可能である。RichReview は会議において、アノテーションを組み合わせることで使うことが可能である。

Miller らは、一週間毎に知り合いのプロジェクトと一緒に作業するペアを組む、PairResearch を行った [5]。PairResearch はメンバーが物事を終わらせる、他人の専門知識を共有することを支援する。ペアの作成は、あるメンバーが手伝って欲しい作業を示し、他のメンバーはその作業にどれだけ貢献できるかを示す。入力された内容から機械的にペアを作成し、作業を行った。

我々は、過去に作成したスライドを利用して、スライドの作成を支援するシステムを試作した [1]。この研究では、検索時にアノテーションを付与し、次の検索時に付与したアノテーションについて検索することを可能にした。PowerPoint マクロとして作成することで、PowerPoint 上でシームレスにスライド検索を行うことを可能にした。

## 6. おわりに

本研究では、多重透明レイヤーをデスクトップ上に重ねて表示する Web ブラウザであるシルエット Web ブラウザを提案した。シルエット Web ブラウザは既存のデスクトップのレイヤー構造にシルエット Web ブラウザを用いてユーザが利用可能なレイヤーを追加した。

シルエット Web ブラウザは、ウィンドウレベルの異なる 4 つのレイヤーを作成する。ユーザは、各レイヤー上に、Web コンテンツを表示可能なフレームを作成することが可能である。フレームは、追跡フレーム、タイムマシンフレーム、スクリーンショットフレームとファイル共有フレームの要素機能を持っている。フレーム間は、情報のインタラクションが可能である。ユーザはフレームの機能を組み合わせて複雑な機能を利用することが可能である。

シルエット Web ブラウザの応用例として、添削システムを挙げた。添削者は、デスクトップ PC 上で、添削範囲を指定する。指定した添削範囲をタブレット端末から添削をする。タブレット端末は、タッチイベントによって操作するため、直感的な操作が可能である。しかし、画面サイズの制約上、書類全体を把握することは困難である。ユーザは、デスクトップ PC で、資料全体から添削範囲を選択する操作を行う。タブレット端末上で、選択した資料に対して、添削を行う。

添削システムの例では、シルエット Web ブラウザは、添削者が利用する端末間と添削者と被添削者間での情報のやりとりをシームレスに行うことを可能にした。SSF を利用することで、端末に依存せずに画面共有を行うことが可能である。添削システムでは、画面共有を行っている端末からシルエットブラウザへの入力を行うことで添削システムを実現した。

#### 参考文献

- [1] 杉山亮弘, 白松俊, 大園忠親, 新谷虎松. "アノテーションを利用したスライド作成支援システムの試作", 第 12 回情報科学技術フォーラム, 2013.
- [2] 片山真也, 合田拓史, 白松俊, 大園忠親, 新谷虎松. "Web 技術に基づく協調型資料添削支援システムの実現", 人工知能学会全国大会論文集, 2013.
- [3] Yoon, Dongwook, et al. "RichReview: blending ink, speech, and gesture to support collaborative document review." Proceedings of the 27th annual ACM symposium on User interface software and technology. ACM, 2014.
- [4] Fourney, Adam, et al. "InterTwine: creating inter-application information scent to support coordinated use of software." Proceedings of the 27th annual ACM symposium on User interface software and technology. ACM, 2014.
- [5] Miller, Robert C, et al. "Pair research: matching people for collaboration, learning, and productivity." Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing. ACM, 2014.