

# Web アプリケーションとして動作する デスクトップマスコットの提案

遠藤崇<sup>†1</sup> 鈴木浩<sup>†1</sup> 速水治夫<sup>†1</sup>

PC のデスクトップにイラストのキャラクターを配置し、画像の連続差し替えや画像周辺のバルーンにセリフを載せて、あたかもそこに喋るキャラクターがいるような感覚を提供するアプリケーションをデスクトップマスコットという。デスクトップマスコットは有志や企業などで開発され、多くは Web でプログラムが公開されている。また、新規にキャラクターを追加できるようになっており、ユーザによるキャラクターデータの追加がよく行われている。しかし、プログラムダウンロードの手間が必要なことや、ユーザが追加したキャラクターデータの評価がデータ作成者へフィードバックされにくいといった問題点があった。そこで著者らはブラウザ上で動作する、キャラクターデータの投稿や共有、評価が可能な Web アプリケーションを開発した。Web アプリケーションとして動作することにより、キャラクターデータの配布に留まらず、フィードバックが可能になった。

## Proposal of Desktop Mascot that Runs as a Web Application

TAKASHI ENDO<sup>†1</sup> HIROSHI SUZUKI<sup>†1</sup>  
HARUO HAYAMI<sup>†1</sup>

Place the illustrations of character to a PC desktop, and put an image in continuous replacement and a words in balloon around the image, desktop mascot as though it were an application that provides a feeling of being a character that speaks in there. Desktop mascot is developed in such as volunteers and companies, many of which are published program on the Web. Also, it has become possible additional new character, additional character data is often done by users. However, there is problems that it require labor of program download, and evaluation of the character data that a user has added is not easily be fed back to a data creator. So authors developed Web application that can be posted and shared and evaluation of character data that runs on the browser. By running as a Web application, not only distribution of character data, became possible feedback.

### 1. はじめに

コンピュータの普及とともに、ユーザが楽しむことを目的としてそれを利用することが多くなっている。その目的は多様化しており、ゲームやコミュニケーション、映像や音楽の鑑賞などに利用されている。それらを実現するためのアプリケーションもよく普及している。

こうした中で取り上げるのは「デスクトップマスコット」アプリケーションである。デスクトップマスコットとは、瀧田航一朗他（2005）[1]によれば、「Microsoft Windows や Mac OS 等の GUI ベースの環境において、主画面(デスクトップ)の片隅に配置される、仮想の外見と人格を与えられたエージェントプログラムである。中にはスケジュール管理やメール確認などの機能を持つものもあるが、基本的には、スクリプト化された疑似的な会話やアニメーションによる仕草などでユーザを楽しませることを目的とする。」

デスクトップマスコットのアプリケーションはある 1 つのキャラクターを表示する場合もあるが、本体のプログラムとキャラクターデータを分離し、任意のキャラクターを追加することが可能なものもある。キャラクターデータは表示に使用される画像や、セリフや振る舞いを記述したテキストまたはスクリプトを、キャラクター単位でまとめた

ものである。キャラクターデータはプログラムに対して複数、追加しておくことも可能である。このとき、本体のプログラムは 1 つのキャラクターデータを読み込んでキャラクターを表示する[2]。読み込まれるキャラクターデータはユーザが指定した設定により決定される。

本体のプログラムの開発者は 1 つ以上のキャラクターデータを追加した状態で、多くの場合 Web 上に配布する[3][4]。このプログラムが任意のキャラクターを追加可能なとき、ユーザは自らキャラクターデータを作成し、それをプログラムに読み込んで表示させることが可能である。データの作成はプログラムの開発者によってあらかじめ公開されたデータの仕様や書式に沿って行われる。データ作成者であるユーザもまた、多くの場合キャラクターデータを Web 上で配布する[5][6]。さらに他のユーザがデータ作成者の作成したキャラクターデータをダウンロードし、自らの環境で表示することが可能である。

これらの一連の流れについて筆者らは 2 点の問題があると考えた。1 点目はインストールの手間である。いずれの行程も Web 上で行われることが多いが、各データ作成者の Web サイトでキャラクターデータが配布されるなど、配布元 Web サイトが統一されていない。またキャラクターデータは単体では動かず、インストールは本体のプログラムのインストールとは別に行う必要がある。2 点目はデータ作成者が作成したデータについてのフィードバックを得られ

<sup>†1</sup> 神奈川工科大学  
Kanagawa Institute of Technology

にくいことである。フィードバックとは作成したキャラクターデータが他のユーザによってどの程度使われているのか、あるいはどの程度好まれているのかといった情報をキャラクターデータ作成者が得ることである。キャラクターデータの配布元 Web サイトは個人や有志による運営のために、ダウンロード数のカウントや投票の機能を持たない低コストの Web サイトを使用していることが多い。また、その機能を有した投稿プラットフォームがない。

そこで本研究ではキャラクターデータの共有とデータ作成者へのフィードバックが可能なアプリケーションを提案する。提案するアプリケーションは従来のアプリケーションと同様の機能を提供し、ユーザを楽しませることを目的としている。

アプリケーションは、Web アプリケーションとして実装する。従来のアプリケーションの基本的な機能であるキャラクターを表示する機能に加えて、キャラクターデータを投稿する機能を備えた。これにより、配布元の統一とインストールの手間の削減、そしてデータ作成者に対するフィードバック手段の構築を実現する。

本研究は、提案アプリケーションの構成と使用方法、評価実験の結果と考察、おわりに、という構成で示す。

## 2. 提案アプリケーション

### (1) アプリケーション構成

アプリケーションは MySQL と PHP, Apache で構築されたサーバと、ブラウザであるクライアントから構成されている。サーバは HTML を生成しクライアントに送信する。クライアントは受信した HTML を整形して表示する。

サーバはデータベース機能と HTML ファイルなどの送受信機能を提供する。データベースには投稿されたキャラクターデータの情報、ユーザのログイン情報、使用状況をテーブルにまとめて記録する。表 1 から表 6 は各テーブルの項目と型を示したものである。ファイルの送受信はクライアントからのリクエストに応じて HTML の生成とクライアントへの送信、キャラクターデータの投稿や評価などのパラメータのクライアントからの受信、データベース操作を行う。

クライアントはサーバから送られた HTML の受信と表示、キャラクターデータの投稿や評価などのパラメータをサーバへ送信する機能を提供する。また、ユーザの操作や表示に用いるインターフェースの機能も持つ。クライアントの画面遷移図を図 1 に示す。ユーザは始めにトップページからユーザ名とパスワードを用いたユーザログインを行う。これらの情報を持たない場合には同ページから新規ユーザ登録が可能である。トップページ以外のページに行くためにはログインを経ないと情報を見ることができない。これはどのユーザがキャラクターデータを投稿し、表示したかをフィードバックや投稿一覧で表示するためである。

ユーザがキャラクターデータに対して表示や評価などを行ったとき、サーバはその日時と連番をデータベースに記録する。表示の場合は、表示のリクエストがクライアントから送られてきた際に、連番と表示開始時刻を記録する。クライアントが別のページに遷移したときは、Ajax を用いてバックグラウンドでサーバに開始時刻記録時の連番を送り、同じ連番と表示終了時刻を記録する。これにより、ユーザがどの程度の時間、表示していたかを把握することが可能である。評価の場合は、評価をした際に、Ajax を用いて評価値をサーバに送り、連番と日時をデータベースに記録する。

表 1 パターンテーブル

Figure 1 Patterns table.

カラム	型	備考
キャラクターID	int(11)	主キー
画像ファイル ID	int(11)	主キー
セリフ	varchar(80)	主キー

表 2 画像ファイルテーブル

Table 2 Image files table.

カラム	型	備考
画像ファイル ID	int(11)	主キー、連番
キャラクターID	int(11)	ユニーク
ファイル名	varchar(80)	ユニーク
使用状態	int(11)	

表 3 キャラクターデータテーブル

Table 3 Character dates table.

カラム	型	備考
キャラクターID	int(11)	主キー、連番
名前	varchar(80)	
ユーザ ID	int(11)	
サムネイル	int(11)	
画像ファイル ID		
説明文	varchar(120)	
公開状態	int(11)	

表 4 ユーザテーブル

Table 4 Users table.

カラム	型	備考
ユーザ ID	int(11)	主キー、連番
ユーザ名	varchar(80)	ユニーク
パスワード	varchar(80)	

表 5 表示状況テーブル

Table 5 Display situations table.

カラム	型	備考
表示 ID	int(11)	連番
ユーザ ID	int(11)	主キー
日付	date	主キー

時間	time	主キー
キャラクターID	int(11)	主キー
表示状態	int(11)	

表 6. 評価状況テーブル

Table 6 Evaluation situations table.

カラム	型	備考
評価 ID	int(11)	連番
ユーザ ID	int(11)	主キー
日付	date	主キー
時間	time	主キー
キャラクターID	int(11)	主キー
評価値	int(11)	

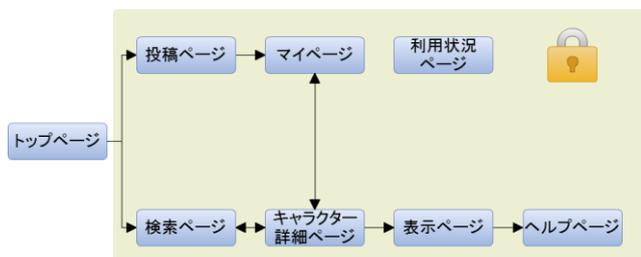


図 1 アプリケーションの画面遷移図

Figure 1 Page transition diagram of application.

## (2) 使用方法

使用方法として 2 通りのユーザを想定している。1 つはキャラクターデータを投稿するユーザ、もう 1 つは投稿されたキャラクターデータを表示するユーザである。

まず、投稿するユーザの使用方法について説明する。前準備としてユーザはキャラクターデータを構成する画像とテキストを用意する必要がある。本アプリケーションでは表示に使用されるいくつかの画像と、画像隣のバルーンでの表示に使用されるいくつかのテキストが必要である。これらは本アプリケーションでは行わず、他のフォトタッチソフトやテキストエディタなどを用いる。画像は表示される表情や動きのパターン数を留意し、画像上のキャラクターの位置はどの画像も固定の位置とする。形式は PNG とし、背景はアルファチャンネルを用いた透過色を使用する。テキストは画像に対応するテキストを用意するが、1 つの画像に対して複数のテキストを対応させることも可能である。次に投稿ページにアクセスし、用意した画像のアップロードを行う。画像を選択し、テキストを入力した後、パターンの追加を行うことで、表示される画像と入力したテキストの組み合わせであるパターンがフォームのリストに追加される。最後にキャラクターデータの名前やサムネイル、説明文の入力を行い、投稿する。この際、リストのパターンがサーバに送られる。他のユーザによる表示や評価があった後、マイページにアクセスすることで、ある特定のユーザによる表示の通知や評価値が表示される。

次に、表示するユーザの使用方法について説明する。ユーザはまず検索ページにアクセスする。このページでは投稿された複数のキャラクターデータがサムネイル、キャラクター名、投稿したユーザ名として表示されている。ユーザはこの中から 1 つを選択すると、キャラクターデータの詳細を表示するページにアクセスする。このページで累計の表示回数と累計の平均評価値を確認でき、評価が行える。また、表示ページへのリンクがあり、表示ページへの遷移も行える。表示はキャラクターデータの持つパターンのうち 1 つをランダムに取り出して、パターンの画像と、隣にバルーンを表示しパターンのテキストの表示を行う。パターンは 6 秒ごとに更新される。評価は 5 段階で行う。短時間の連続評価はできず、ユーザは 1 つのキャラクターデータに対して 1 日 1 回の評価が可能である。

本アプリケーションでは最初にユーザ登録したユーザを管理者と想定し、管理者のみがアクセス可能な各ユーザによる使用状況を確認するページがある。このページはユーザが行ったキャラクターデータの表示や評価の内容を、3 つの表で表す。表からはあるユーザがあるキャラクターに対して、いつ使用されたか、どの程度の期間で使用されたか、いついくつの評価があったかがそれぞれ分かるようになっている。また、アクセスした際にサーバに CSV ファイルが保存され、表計算ソフトウェアなど他のアプリケーションで使用状況のデータを使用することも可能である。

## 3. 評価実験

### 3.1 結果

提案アプリケーションによるインストールの手間の削減とフィードバック手段の構築が実現できているかを確認するため、20 代の大学生、大学院生 7 名を対象に評価実験を行った。被験者のうち、デスクトップマスコットを使用したことがある人は 2 名で、残りの 5 名は今回初めての使用であった。実験では、実際にアプリケーションを使用してもらい、使用後にアンケート評価を行った。アンケート項目は次の 4 つである。項目はそれぞれ 5 段階評価とし、最後の項目のみ自由記述となっている。また、フィードバックに関する項目は 6 名のうち、キャラクターデータの投稿を行った 4 名のみが回答している。

- 対象アプリケーションでのキャラクターの表示に満足できたか
- 対象アプリケーションは使用準備 (ダウンロードとインストール、キャラクターデータの追加) の手間は減らせたか
- マイページのお知らせに表示されるフィードバックに満足できましたか
- 他に必要と思う機能があればご記入ください

表 7 は項目別の評価の結果である。表中の値は人数を示す。括弧内はこのうちデスクトップマスコットを使用したこと

があると回答した人の人数である。自由記述では次のコメントがあった。

- 既存のデスクトップ・マスコットのほうがよりキャラクターらしいものを作成・表示することができる
- キャラクターの評価がやりにくい。
- よく使われる、よく好まれるの違いが分かりにくい。
- 音声機能
- 単純にテキストチャとセリフを対応付けているだけなので人工無能的な機能が欲しい

ユーザ毎にアプリケーションの表示時間を使用状況として取得しており、これを図2に示す。キャラクターデータの表示は63件あった。このうち、56件で表示時間を取得することができた。その結果、1分以上の表示は9件、10分以上の表示は3件となった。また、平均表示時間は4分34秒であった。多数回利用した4ユーザは1回の表示時間はほぼ1分以下であった。一方利用回数の少ないユーザは1回に10分以上表示した3ユーザと表示時間が8秒以下の2ユーザであった。キャラクターデータの評価は17件あった。すべてのキャラクターデータの平均の評価値は4.18であった。各ユーザは2~3のキャラクターデータに対して評価を行っていた。

表7 アンケート評価の結果

Table 7 Result of evaluation questionnaire.

	5	4	3	2	1	平均
表示の満足度	0	2	3	1(1)	1(1)	2.86
使用準備の手間	0	3	3(1)	1(1)	0	3.29
フィードバック	0	2(1)	0	2(1)	0	3.00

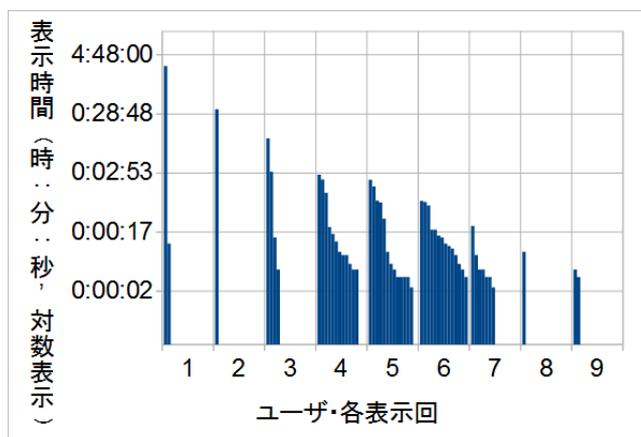


図2 ユーザごとの各表示回のキャラクターデータの表示時間

Figure 2 display period of each display of character data per user.

### 3.2 考察

アンケート評価と使用状況の分析からある程度の評価を得られた。1点目の問題点として提示した使用準備の手間の削減については、評価アンケートの結果からある程度は解決できたと考えている。しかし、評価アンケートや使用状況について結果が芳しくなかった。要因として2点考えられる。1点目は本アプリケーションのキャラクターデータ表示機能がデスクトップマスコットとしての基本的な機能のみで実装されたことである。評価アンケートの自由記述に多く見られるように、楽しませる目的としては不十分であったと考えられる。2点目は操作や表記といったユーザインターフェースの問題である。評価実験当時、連続評価の際に表示がない、「表示する」と「使う」や「評価」と「好まれる」の表記が統一されていない状況であった。こうした点が影響したものと考えられる。この2点を改善して再度評価を行うことで、評価の向上に繋がると考えられる。

### 4. おわりに

デスクトップマスコットのキャラクターデータの使用準備の手間の削減や、キャラクターデータ作成者が使用や評価といったフィードバックを得られやすくするために、本研究ではWebアプリケーションとして動作するデスクトップマスコットの提案を行った。これにより、キャラクターデータの表示や投稿の手間を減らし、さらに他のユーザによる表示の通知や評価を行うことが可能になった。評価実験では本アプリケーションを実際に使ってもらい、問題のある程度解決できたことを確認できた。一方で表示機能は楽しませる目的を十分に果たせていない、フィードバックの評価が高くないといった問題があり、今後の課題となる。よりキャラクターに近い表示の表現として人口無能の実装を考えている。

### 参考文献

- 1) 瀧田航一朗, 萩原将文: 感性情報の学習と収集を行う対話型デスクトップマスコットの提案 感性工学研究論文集, Vol.5, No.3, pp.81-86 (2005)
- 2) 松浦健一郎, 司ゆき: デスクトップマスコットを作ろう!! , ソフトバンクパブリッシング, 2003
- 3) 何か  
<http://usada.sakura.vg/>
- 4) Apricot for Windows  
<http://www.apricotan.net/apricot-for-windows/>
- 5) GHOST TOWN  
<http://ghosttown.mikage.jp/>
- 6) SiReFaSo  
<http://sirefaso.appspot.com/>