

【特許請求の範囲】

【請求項 1】

通信ネットワークを介して接続される複数の端末に仮想コミュニティ環境を提供するコミュニティ環境提供システムにおいて、

各々の前記端末からオブジェクトの制御情報を受信する第 1 受信手段、および

前記制御情報に基づく表現をする前記オブジェクトを前記仮想コミュニティ環境に再現する再現手段を備えることを特徴とする、コミュニティ環境提供システム。

【請求項 2】

前記制御情報は、前記仮想コミュニティ環境における前記オブジェクトの再現位置情報、前記オブジェクトの移動先を示す移動先情報、前記仮想コミュニティ環境の所定位置を示唆する指示情報、前記オブジェクトの振る舞いに関する振る舞い情報、および前記オブジェクトによる発話に関する発話情報の少なくとも 1 つを含む、請求項 1 記載のコミュニティ環境提供システム。 10

【請求項 3】

前記制御情報は前記オブジェクトの表現に関連して出力されるマルチメディアコンテンツを含む、請求項 1 または 2 記載のコミュニティ環境提供システム。

【請求項 4】

前記制御情報を記録する記録手段、

前記端末から前記制御情報の送信要求を受信する第 2 受信手段、および

前記記録手段に記録された前記制御情報を前記送信要求に基づいて返送する返送手段をさらに備える、請求項 1 ないし 3 のいずれかに記載のコミュニティ環境提供システム。 20

【請求項 5】

通信ネットワークに接続された情報処理装置によって提供される仮想コミュニティ環境でオブジェクトに所望の表現をさせるオブジェクト表現制御装置であって、

前記オブジェクトの表現態様の入力を受け付ける受付手段、

前記受付手段によって受け付けた前記表現態様に基づいて前記オブジェクトの制御情報を作成する作成手段、および

前記制御情報を前記通信ネットワークを通じて前記情報処理装置に送信する送信手段を備える、オブジェクト表現制御装置。 30

【請求項 6】

前記受付手段は複数の表現項目を有する表現メニューを表示して所望の表現項目の選択を受け付ける第 1 受付手段を含み、

前記作成手段は前記所望の表現項目に基づいて前記制御情報を作成する第 1 作成手段を含む、請求項 5 記載のオブジェクト表現制御装置。

【請求項 7】

前記表現メニューは、複数の振る舞い項目を有する振る舞いメニュー、および複数の移動先項目を有する移動先メニューの少なくとも一方を含む、請求項 6 記載のオブジェクト表現制御装置。

【請求項 8】

前記受付手段は前記オブジェクトが発する台詞の入力欄を表示して所望の台詞の入力を受け付ける第 2 受付手段を含み、 40

前記作成手段は前記所望の台詞に基づいて前記制御情報を作成する第 2 作成手段を含む、請求項 5 ないし 7 のいずれかに記載のオブジェクト表現制御装置。

【請求項 9】

前記受付手段は前記オブジェクトが引用表現するマルチメディアコンテンツの発信元情報の入力欄を表示して所望の発信元情報の入力を受け付ける第 3 受付手段を含み、

前記作成手段は前記所望の発信元情報に基づいて前記制御情報を作成する第 3 作成手段を含む、請求項 5 ないし 8 のいずれかに記載のオブジェクト表現制御装置。

【請求項 10】

前記制御情報は前記表現態様の種類を示す識別子を含む、請求項 5 ないし 9 のいずれか 50

に記載のオブジェクト表現制御装置。

【請求項 11】

通信ネットワークに接続された情報処理装置によって提供される仮想コミュニティ空間上でオブジェクトに所望の表現をさせるオブジェクト表現制御装置の制御プログラムであって、

前記オブジェクトの表現態様の入力を受け付けるステップ、

前記受付手段によって受け付けた前記表現態様に基づいて前記オブジェクトの制御情報を作成するステップ、および

前記制御情報を前記通信ネットワークを通じて前記情報処理装置に送信するステップを備える、オブジェクト表現制御装置の制御プログラム。

10

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、コミュニティ環境提供システムに関し、特にたとえば、通信ネットワークを介して接続される複数の端末に仮想コミュニティ環境を提供する、コミュニティ環境提供システムに関する。

【0002】

この発明はまた、オブジェクト表現制御装置またはオブジェクト表現制御プログラムに関し、特にたとえば、通信ネットワークに接続された情報処理装置によって提供される仮想コミュニティ空間上でキャラクタエージェントのようなオブジェクトに所望の表現をさせる、オブジェクト表現制御装置またはその制御プログラムに関する。

20

【背景技術】

【0003】

従来、ネット上に形成される BBS (Bulletin Board System: 電子掲示板システム) などのこの種のコミュニティ環境提供システムでは、情報を交換するための媒体はテキストデータであった。また、URL (Uniform Resource Locator) をテキストデータで示すことにより、提供したい情報が掲載されたウェブページを教えるという方法もあった。

【発明の開示】

【発明が解決しようとする課題】

【0004】

しかし、テキストデータでは表現できる情報内容が乏しく、また情報交換が発展するにつれて提供された情報がどの情報に対する情報提供 (意見, 感想など) であるかが不明確になるなどし、相手に伝えられる情報の質および正確さに限界があった。また、テキストデータのみによる表現では情緒が十分に表現されず情報提供者の意図が十分に伝わらない、もしくは誤解されるという大きな問題もあった。

30

【0005】

URL でウェブページを教える方法を用いても、ウェブページによって提供された情報について感想を述べたり、ウェブページによって共有した情報を基に会話を発展させる場合に、情報源となったウェブページの文章および映像データなどの情報自体がそのコミュニティ上に存在しないため情報の伝達に不自由するという問題があった。

40

【0006】

また、キャラクタエージェントなどのオブジェクトを用いると、オペレータの意図を正確に表現できるが、オブジェクトによる表現では表現の制御に手間がかかるという問題がある。さらに、このような情報内容の計算機による自動解析や情報検索において、何らの構造化もされていないテキストデータでは、その精度に自ずと限界がある。

【0007】

それゆえに、この発明の主たる目的は、より質が高くより正確な情報交換がスムーズに行える、コミュニティ環境提供システムを提供することである。

【0008】

この発明の他の目的は、コミュニティ環境に登場するオブジェクトの表現を容易に制御

50

することができる、オブジェクト表現制御装置またはその制御プログラムを提供することである。

【課題を解決するための手段】

【0009】

第1の発明は、通信ネットワークを介して接続される複数の端末に仮想コミュニティ環境を提供するコミュニティ環境提供システムにおいて、各々の端末からオブジェクトの制御情報を受信する第1受信手段、および制御情報に基づく表現をするオブジェクトを仮想コミュニティ環境に再現する再現手段を備えることを特徴とする、コミュニティ環境提供システムである。

【0010】

第2の発明は、通信ネットワークに接続された情報処理装置によって提供される仮想コミュニティ環境でオブジェクトに所望の表現をさせるオブジェクト表現制御装置であって、オブジェクトの表現態様の入力を受け付ける受付手段、受付手段によって受け付けた表現態様に基づいてオブジェクトの制御情報を作成する作成手段、および制御情報を通信ネットワークを通じて情報処理装置に送信する送信手段を備える、オブジェクト表現制御装置である。

【0011】

第3の発明は、通信ネットワークに接続された情報処理装置によって提供される仮想コミュニティ空間上でオブジェクトに所望の表現をさせるオブジェクト表現制御装置の制御プログラムであって、オブジェクトの表現態様の入力を受け付けるステップ、受付手段によって受け付けた表現態様に基づいてオブジェクトの制御情報を作成するステップ、および制御情報を通信ネットワークを通じて情報処理装置に送信するステップを備える、オブジェクト表現制御装置の制御プログラムである。

【0012】

第1の発明においては、仮想コミュニティ環境が通信ネットワークに接続された複数の端末に提供される。第1受信手段が各々の端末からオブジェクトの制御情報を受信すると、受信した制御情報に基づく表現をするオブジェクトが再現手段によって仮想コミュニティ環境に再現される。

【0013】

制御情報は、好ましくは、仮想コミュニティ環境におけるオブジェクトの再現位置情報、オブジェクトの移動先を示す移動先情報、仮想コミュニティ環境の所定位置を示唆する指示情報、オブジェクトの振る舞いに関する振る舞い情報、およびオブジェクトによる発話に関する発話情報の少なくとも1つを含む。

【0014】

制御情報は、オブジェクトの表現に関連して出力されるマルチメディアコンテンツを含んでもよい。

【0015】

制御情報を記録し、端末から制御情報の送信要求を受信したときに当該制御情報を要求元に返送する場合、過去に受信した制御情報に基づいてオブジェクトを表現させることができる。

【0016】

第2の発明においては、情報処理装置によって通信ネットワークを通して提供される仮想コミュニティ環境でオブジェクトに所望の表現をさせるとき、受付手段は、オブジェクトの表現態様の入力を受け付ける。オブジェクトの制御情報は、受け付けた表現態様に基づいて作成手段によって作成され、作成された制御情報は、送信手段によって通信ネットワークを通じて情報処理装置に送信される。

【0017】

複数の表現項目を有する表現メニューを表示して所望の表現項目の選択を受け付ける場合、制御情報は、受け付けた所望の表現項目に基づいて作成される。表現メニューは、好ましくは、複数の振る舞い項目を有する振る舞いメニュー、および複数の移動先項目を有

10

20

30

40

50

する移動先メニューの少なくとも一方を含む。

【0018】

オブジェクトが発する台詞の入力欄を表示して所望の台詞の入力を受け付ける場合、制御情報は受け付けた所望の台詞に基づいて作成される。

【0019】

オブジェクトが引用表現するマルチメディアコンテンツの発信元情報の入力欄を表示して所望の発信元情報の入力を受け付ける場合、制御情報は、受け付けた所望の発信元情報に基づいて作成される。

【0020】

作成手段によって作成される制御情報がオブジェクトの表現態様の種類を示す識別子を有する場合、情報処理装置側では、受信した制御情報を識別子に基づいて検索したり解析したりすることができる。つまり、GUIによりユーザに意識させることなく、表現の手段ごとの多次元な識別子を表現情報内に付加することができる。このように多次元な識別子により構造化された情報は、計算機による意味解析や情報検索の質と精度の向上に寄与し、コミュニティ内外に対する情報の再利用に役立たせることができる。

【0021】

第3の発明においては、情報処理装置によって通信ネットワークを通して提供される仮想コミュニティ環境でオブジェクトに所望の表現をさせるとき、まずオブジェクトの表現態様の入力を受け付けられる。オブジェクトの制御情報は受け付けられた表現態様に基づいて作成され、作成された制御情報は通信ネットワークを通じて情報処理装置に送信される。

【発明の効果】

【0022】

第1の発明によれば、各々の端末から受信した制御情報に基づいて仮想コミュニティ環境のオブジェクトが表現をするため、より質が高くより正確な情報の交換をスムーズに行うことができる。

【0023】

第2の発明または第3の発明によれば、受け付けた表現態様に基づいてオブジェクトの制御情報を作成するようにしたため、オブジェクトの表現を容易に制御することができる。

【0024】

この発明の上述の目的、その他の目的、特徴および利点は、図面を参照して行う以下の実施例の詳細な説明から一層明らかとなろう。

【発明を実施するための最良の形態】

【0025】

図1を参照して、この実施例においてはサーバ10がコミュニティ環境提供システムを形成する。サーバ10はインターネット60などのネットを介して複数のクライアント30と接続可能になっている。以下において、クライアント30のオペレータ（以下、単に「オペレータ」と表記する）をユーザとも呼ぶ。この実施例のコミュニティ環境提供システムでは、複数のユーザがクライアント30からサーバ10に接続し、サーバ10が提供するコミュニティ環境において互いの情報を交換し、また自己表現を行う。

【0026】

サーバ10は図2に示すように構成され、入力装置としてのキーボード（マウスなども含む）14、バッファ16を介した通信回路18、ディスプレイ20、コミュニティ環境を形成するデータが記録されたHD（Hard Disc）22およびメモリ24がバス26にそれぞれ接続され、これらは同じくバス26に接続されたCPU12によって制御される。

【0027】

また、クライアント30は図3に示すようにサーバ10と同様の構成であり、バス46にマウスなども含む入力装置としてのキーボード34、バッファ36を介した通信回路38、ディスプレイ40、HD42およびメモリ44が接続され、これらは同じくバス46

10

20

30

40

50

に接続されたCPU32によって制御される。

【0028】

この実施例のコミュニティ環境提供システムでは、複数のユーザがサーバ10にアクセスして情報交換、自己表現などを行う。情報交換の場合は趣味や話題の分野別に分かれており、その分野の1つ1つを「サークル」と呼ぶ。たとえば、写真についてのサークル、音楽についてのサークル、映画についてのサークルなどである。また、サークルにおいては、その分野についての「話題」がユーザによって提供され、複数のユーザがその話題について情報交換を行う。たとえば、写真についてのサークルの話題としては、「空の写真を集めたホームページを見つけました」などである。

【0029】

各ユーザはサークルにおいてユーザを代理するインタフェースエージェント（キャラクタエージェント）として、図4に示すようなキャラクタ（キャラクタ1）をサーバ10にそれぞれ登録する。そしてユーザはこのキャラクタを介してサークルで活動する。各サークルはウェブページ（ディスプレイ上の部分領域または全領域）で表現され、ユーザはウェブページに表示されるキャラクタを操作して、発言、ウェブページ上の移動、アニメーションによるアクション（リアクション）などを行う。なお、以下において発言およびウェブページ上の移動を便宜上「コメント」と呼ぶこともある。また、ユーザは所望のウェブページおよび画像、動画などをサークル（話題）のウェブページに表示して、話題に参加している他のユーザに提示することもでき、これらの動作も「コメント」の概念に含める。一人のユーザがコメントを行って話題や情報を提供すると、他のユーザがその話題や情報に対するコメントやリアクションを行う。そして、そのコメントおよびリアクションに対し、さらに他のユーザがコメントおよびリアクションを行う。このようにして従来のBBSのように情報交換が行われる。この情報交換の過程はサーバ10のHD22にスレッドデータ（詳細は後述する）として記録される。そして、後から話題に参加したユーザはスレッドデータを基にその話題において行われた情報交換のやり取りを再現することができる。後から参加したユーザはその話題で行われている情報交換に興味があれば、さらに自分の情報を提示したり感想を述べたりすることができる。

【0030】

サーバ10のHD22は図7に示すようにユーザ情報データベース70、キャラクタデータベース74、サークルデータベース80、ウェブページデータベース84などから構成されており、各話題のスレッドデータはサークルデータベース80に記録される。サークルデータベース80はさらにサークル別の複数のスレッドデータベース82に分かれている。このスレッドデータベース82は図8に詳しく示すように構成されている。すなわち、スレッドデータベース82の内部は大きくは話題ごとに、たとえばフォルダによって区切られており、話題フォルダ90の内部には、その話題に登場するキャラクタの情報が収められた登場キャラクタフォルダ92、情報交換の経過を「場面」の単位で記録する場面フォルダ94が含まれている。ここで、場面とは話題を構成する単位であり、場面は場面1、場面2、場面3...というように情報交換の経過順に連鎖している。この場面のつながりのことを「スレッドデータ」と呼ぶ。各場面フォルダ94には、ユーザのコメントが記録された一つのコメントスクリプトと、コメントに対して他のユーザが行ったリアクションが記録された複数のリアクションスクリプト（コメントに対してリアクションがない場合も当然あり得る）が保存される。このとき、各々のスクリプトは、時系列的に構造化された状態で保存される。コメントおよびリアクションの詳細については後述する。

【0031】

ユーザ情報データベース70にはコミュニティ環境提供システムを利用するユーザのユーザID、パスワード、ユーザの個人情報などを含む複数のユーザ情報72...72がユーザ別に記録されている。また、ユーザ情報72にはユーザによって登録されたインタフェースエージェント（キャラクタ）の種類などを示すキャラクタ情報（たとえばキャラクタID）が含まれる。

【0032】

10

20

30

40

50

図 7 に戻って、キャラクタデータベース 7 4 はデフォルトキャラクタデータベース 7 6 , オリジナルキャラクタデータベース 7 8 などを含んでいる。デフォルトキャラクタデータベース 7 6 には、予めサーバ 1 0 に設定されたキャラクタのデータが記録されており、各キャラクタはキャラクタ ID によって識別される。ユーザはデフォルトキャラクタデータベース 7 6 に記録されているキャラクタを選択して自分のインタフェースエージェントとして使用できる。また、ユーザは自分の作成したキャラクタをインタフェースエージェントとして使用することもでき、ユーザのオリジナルキャラクタはオリジナルキャラクタデータベース 7 8 に記録される。

【 0 0 3 3 】

なお、インタフェースエージェントとしてのキャラクタは、クライアント 3 0 の H D 4 2 に記録されている専用のプログラムによってウェブページ上に表示される。したがって、デフォルトキャラクタデータベース 7 6 およびオリジナルキャラクタデータベース 7 8 に登録されているキャラクタデータはこのプログラムの仕様に従ったデータ形式である。キャラクタを表示するプログラムとしては、たとえば Microsoft Agent が利用できる。

【 0 0 3 4 】

また、ウェブページデータベース 8 4 には、各ウェブページを生成するための雛型ウェブページを記録した雛型データベース 8 6 および生成されたウェブページを記録するサークルページデータベース 8 8 などが含まれている。

【 0 0 3 5 】

以下、この実施例でのコミュニティ環境提供システムをサーバ 1 0 の C P U 1 2 およびクライアント 3 0 の C P U 3 2 が行う処理を示すことによって説明する。C P U 1 2 および C P U 3 2 は、図 1 3 から図 2 8 に示すフロー図に従って処理を行う。なお、図 1 3 から図 2 8 のフロー図はサーバ 1 0 側の処理とクライアント 3 0 側の処理を連携させた形式で示している。

【 0 0 3 6 】

まず、図 1 3 のステップ S 1 においてクライアント 3 0 の C P U 3 2 (以下クライアント 3 0 と記述する) はサーバ 1 0 へのアクセス要求をサーバ 1 0 に送信する。サーバ 1 0 の C P U 1 2 (以下サーバ 1 0 と記述する) はステップ S 3 においてアクセス要求を受信してクライアント 3 0 との接続を確立する。そして、コミュニティ環境提供システムにログインするためのログインページ(ウェブページ)をステップ S 5 で送信する。クライアント 3 0 はログインページを受信し、図 9 に示すようなログインページをステップ S 7 でクライアント 3 0 のディスプレイ 4 0 に表示する。オペレータは、コミュニティの会員であればユーザ ID およびパスワードをユーザ ID 入力欄 1 0 0 およびパスワード入力欄 1 0 2 にそれぞれ入力し、OK ボタン 1 0 4 を選択する。一方、コミュニティの会員でなければ会員登録ボタン 1 0 6 を選択する。ステップ S 9 において、選択されたのが OK ボタン 1 0 4 であるか会員登録ボタン 1 0 6 であるかを識別して会員であるかどうか判断する。

【 0 0 3 7 】

会員であるときにはステップ S 1 3 においてユーザ ID およびパスワードをユーザ ID 入力欄 1 0 0 およびパスワード入力欄 1 0 2 から取得し、ステップ S 1 5 においてユーザ ID およびパスワードをサーバ 1 0 に送信する。サーバ 1 0 ではユーザ ID およびパスワードを受信し、ステップ S 1 9 においてユーザ ID およびパスワードの認証処理を行う。認証処理はユーザ ID およびパスワードがユーザ情報データベース 7 0 のユーザ情報 7 2 に記録されているかどうかを確認することによって行う。認証しないときにはステップ S 5 に戻り再びログインページをクライアント 3 0 に送信する。認証するときには図 1 6 のステップ S 5 1 においてコミュニティの入り口ページを送信する。

【 0 0 3 8 】

一方、ステップ S 9 において会員でないと判断すると、ステップ S 1 1 において会員登録ページ要求をサーバ 1 0 に送信する。サーバ 1 0 では、会員登録ページ要求を受信して図 1 4 のステップ S 2 1 で会員登録ページ(図示せず)をクライアント 3 0 に送信する。

クライアント 30 では会員登録ページを受信してステップ S 23 でディスプレイ 40 に表示する。オペレータは会員登録ページに対して会員登録に必要な情報を入力する。そして決定ボタン（図示せず）を選択するとステップ S 25 においてユーザ ID、パスワードなどのユーザ情報を会員登録ページから取得してバッファ 36 に格納する。ユーザは会員登録のときに、インタフェースエージェントとしてのキャラクタにデフォルトキャラクタを登録するかオリジナルキャラクタを登録するかのどちらかの登録方法を選ぶことができる。そして、その決定した登録方法に関する情報はユーザ情報に含まれる。

【0039】

ステップ S 27 において、登録方法に関する情報に基づいてオリジナルキャラクタを登録するかどうか判断する。オリジナルキャラクタを登録すると判断するとステップ S 29 10
でオリジナルキャラクタを取り込むためのウィンド（図示せず）をディスプレイ 40 に表示する。オペレータはウィンドに表示される指示に従ってユーザのオリジナルキャラクタを登録する。ステップ S 31 においてキャラクタの登録が完了したかどうか、つまりユーザがウィンドの決定ボタンを選択したかどうか判断する。登録が完了したと判断するとステップ S 33 でキャラクタデータをバッファ 36 に格納し、図 15 のステップ S 45 でバッファ 36 に格納されたユーザ情報およびキャラクタデータをサーバ 10 に送信する。サーバ 10 では、ユーザ情報およびキャラクタデータを受信してステップ S 47 でユーザ登録を行う。ユーザ登録では、キャラクタデータにキャラクタ ID を割り振り、ユーザ情報とキャラクタ ID とをユーザ情報データベース 70（図 7 参照）に記録する。そして、キャラクタデータをキャラクタデータベース 74 のオリジナルキャラクタデータベース 78 20
に記録する。

【0040】

一方、ステップ S 27 においてオリジナルキャラクタの登録でないと判断すると、ステップ S 35 でデフォルトキャラクタ選択メニュー要求をサーバ 10 に送信する。サーバ 10 ではデフォルトキャラクタ選択メニュー要求を受信してデフォルトキャラクタ選択メニューをクライアント 30 に送信する（ステップ S 37）。クライアント 30 では、ステップ S 39 においてデフォルトキャラクタ選択メニューウィンド（図示せず）をディスプレイ 40 に表示する。オペレータはウィンドに表示される指示に従って登録するデフォルトキャラクタを決定し、ウィンドの決定ボタンを選択する。決定ボタンが選択されるとステップ S 41（図 15 参照）においてキャラクタの選択ありと判断し、選択したキャラクタ 30
のキャラクタ ID をステップ S 43 でウィンドから取得してバッファ 36 に格納する。そしてステップ S 45 において、バッファ 36 に格納されているユーザ情報およびキャラクタ ID をサーバ 10 に送信する。サーバ 10 ではステップ S 47 でユーザ情報およびキャラクタ ID を受信してユーザ登録を行う。ユーザ登録ではユーザ情報とキャラクタ ID とをユーザ情報データベース 70 に記録する。

【0041】

サーバ 10 はステップ S 19（図 13 参照）でのユーザ認証およびステップ S 47（図 15 参照）でのユーザ登録が終了した後に、図 16 のステップ S 51 においてキャラクタデータおよびコミュニティの入り口ページ（ウェブページ）をクライアント 30 に送信する。クライアント 30 はキャラクタデータおよびコミュニティの入り口ページ（図示せず） 40
を受信してステップ S 53 でコミュニティの入り口ページおよびキャラクタをディスプレイ 40 に表示する。キャラクタの詳細については後述するが、ここで表示されたユーザのキャラクタはウェブページ上に常に表示される。コミュニティの入り口ページではウェブページ上に設けられたボタンを選択することによって、新規サークルを作成するか既存のサークルに参加するかを選択することができる。

【0042】

新規サークルを作成するときには、コミュニティの入り口ページに設けられている「新規サークルの作成」ボタンを選択する。するとステップ S 55 において新規サークルの作成と判断してステップ S 63 でサークル登録ページの要求をサーバ 10 に送信する。

【0043】

サーバ 10 はサークル登録ページ要求を受信してステップ S 6 5 においてサークル登録ページをクライアント 30 に送信する。

【 0 0 4 4 】

クライアント 30 はサークル登録ページを受信し、ステップ S 6 7 においてサークル登録ページ（図示せず）をディスプレイ 40 に表示する。オペレータはサークル登録ページに設けられた入力欄に登録サークルの情報を入力し、完了ボタンを選択する。完了ボタンを選択するとステップ S 6 9 において情報入力完了と判断し、ステップ S 7 1 で、入力されたサークル情報を取り込んでバッファ 36 に格納する。そしてステップ S 7 3 においてバッファ 36 に格納されているサークル情報をサーバ 10 に送信する。

【 0 0 4 5 】

サーバ 10 はサークル情報を受信してステップ S 7 5 においてサークルの登録を行う。サークルの登録では、サークルデータベース 80（図 7 参照）に当該サークルのスレッドデータベース 82 を作成する。そして、ステップ S 7 7 において当該サークルのサークルページをクライアント 30 に送信する。サークルページはウェブページデータベース 84 の雛型データベース 86 に予め設けられたサークルページの雛型にサークル情報によって与えられた情報を埋め込むことによって生成される。そして生成されたサークルページはサークルページデータベース 88 に記録される。

【 0 0 4 6 】

一方、ステップ S 5 5 において既存のサークルを選択するときには、コミュニティの入り口ページに設けられているサークル名のボタンを選択する。サークル名のボタンとは、たとえば「写真サークル」、「音楽サークル」、「映画サークル」などと表示されたボタンである。オペレータがこれらのサークル名のボタンを選択するとステップ S 5 5 において新規サークル作成でないと判断し、選択された既存サークルをステップ S 5 7 で特定する。サークルは、各サークルに割り振られたサークル ID によって特定される。そして選択されたサークルのページ要求をステップ S 5 9 において送信する。サーバ 10 はサークルページ要求を受信し、選択されたサークルのウェブページをウェブページデータベース 84 のサークルページデータベースから取得してクライアント 30（ステップ S 6 1）に送信する。

【 0 0 4 7 】

次に図 17 においてクライアント 30 がサークルページを受信すると、ステップ S 8 1 において図 4 に示すようなサークルページをディスプレイ 40 に表示する。図 4 のサークルページは既存のサークルページである。ここで、これまでサークルページと呼んできたウェブページは図 4 に示すようにアイコンフレーム、スレッドページおよびサークルページを含む構成となっている。そしてサークルページ上にはユーザが登録したキャラクタ（キャラクタ 1）が引き続き表示される。つぎにステップ S 8 3 においてウェブページの HTML に埋め込まれているスクリプト（詳細については後述）に基づいてキャラクタの動作を操作する。スクリプトに基づく操作によって、キャラクタは図 4 に示すようにサークルで展開されているそれぞれの話題について解説を行う。キャラクタはサークルページ上を移動して各話題の話題名（話題ボタン 120）を指差しながら吹き出し 110 を用いて解説を行う。

【 0 0 4 8 】

オペレータはこのサークルに新たな話題を提供し、もしくは既存の話題に参加することができる。新規話題を提供するときにはアイコンフレーム（図 4 参照）に設けられた新規話題作成ボタン 112 を選択する。また、既存の話題の内容を再現するためにはスレッドページに表示されている話題ボタン 120（図 4 の例では話題 1～話題 3）のいずれかを選択する。話題は「場面（コメントおよびリアクション）」毎に再現され、次の場面ボタン 118 を選択すると次の場面が再現される。そして再現された話題に対してコメントを追加するときにはコメント作成ボタン 114 を選択し、再現された場面にリアクションを追加するときにはリアクション作成ボタン 116 を選択する。なお、コメント作成ボタン 114 は話題に含まれるすべての場面が再現された後に有効となる。また、リアクション

10

20

30

40

50

作成の処理は説明の都合上、後述するステップ S 1 0 7 における「話題再現」の処理の一部として説明する。

【 0 0 4 9 】

これらのいずれかの操作を行うと、クライアント 3 0 ではステップ S 8 5 において操作ありと判断し、ステップ S 8 7 , ステップ S 9 1 , ステップ S 9 5 およびステップ S 9 9 においてそれぞれの操作の内容を判断する。

【 0 0 5 0 】

ステップ S 8 7 において新規話題作成ボタン 1 1 2 が選択されたと判断すると、ステップ S 8 9 において「新規話題作成」の処理を行う。「新規話題作成」の処理は図 1 8 および図 2 0 から図 2 2 に示すフロー図の手順に従って実行される。「新規話題」の処理は、まず図 1 8 のステップ S 1 0 1 において、図 1 0 に示すような新規話題編集ウィンド 1 5 0 をディスプレイ 4 0 のサークルページ上に表示する。そしてステップ S 1 0 3 においてスイッチ L v - S W に「 0 」を設定し、ステップ S 1 0 5 で「スクリプトファイル生成」の処理を行う。なお、L v - S W は「新規話題作成」, 「コメント作成」および「リアクション作成」のいずれの処理を行うかを判断するためのスイッチである。

10

【 0 0 5 1 】

図 2 0 および図 2 1 のフロー図に示す処理は、図 1 8 の「スクリプトファイル生成」の詳細を示しており、「新規話題作成」, 後述する「コメント作成」および「リアクション作成」の各処理に共通である。この「スクリプトファイル生成」の処理では、図 1 0 に示す新規話題編集用ウィンド 1 5 0 に対するユーザの操作に基づいて新規話題作成を行う。なお、新規話題作成には最初のコメント作成が含まれる。

20

【 0 0 5 2 】

まず、図 2 0 のステップ S 1 3 1 で、作成するスクリプトファイルを新規にオープンする。そしてステップ S 1 3 3 において L v - S W が 2 であるか、つまり「リアクション作成」であるかどうか判断する。現在 L v - S W に設定されている値は「 0 」であるのでステップ S 1 3 5 に進む。

【 0 0 5 3 】

ユーザは図 1 0 のウィンドを操作して当該話題の最初のコメントを G U I 形式で作成する。既に述べたようにコメントには台詞だけでなく、画像などのローカルファイルの表示なども含まれる。ユーザがローカルファイルボタン 1 7 0 を選択すると、ローカルファイルを取り込むためのウィンド（図示せず）が表示される。ユーザが予め用意しておいた画像ファイルをウィンドの指示に従って指定することによって画像ファイルを選択する。このとき画像ファイルのサークルページ上の位置も指定するようにする。ローカルファイルの選択が終了するとステップ S 1 5 1 においてローカルファイルの取り込みであると判断し、選択されたローカルファイルをステップ S 1 5 3 でメモリ 4 4 に取り込む。

30

【 0 0 5 4 】

そしてステップ S 1 3 9 において、サークルページ上にローカルファイルを表示させるスクリプトを生成し、生成したスクリプトをステップ S 1 4 1 においてスクリプトファイルに追加する。取り込んだローカルファイルのファイル名がたとえば“ J P E G . 0 0 0 1 ”である場合、ステップ S 1 3 9 で作成されるスクリプトは、“ < # r e f e r > J P E G . 0 0 0 1 ”と記述され、このようなスクリプトが時系列構造で追加される。ステップ S 1 4 1 の処理を終えるとステップ S 1 3 3 に戻り、再び L v - S W の値が 2 であるかどうか判断する。

40

【 0 0 5 5 】

ユーザがプルダウンボタン 1 6 0 （図 1 0 参照）を選択すると移動メニュー表示部 1 5 6 に、キャラクタが移動可能な移動場所の一覧がプルダウンメニュー（図示せず）として表示される。移動場所一覧から移動場所を選択することによってキャラクタを移動させる位置を決定する。移動場所一覧に表示される内容としては、たとえば「画像 1 の右側」, 「キャラクタ 2 の前」, 「サークルページの右隅」などが考えられる。さらに、プルダウンメニューを使用しない場合には、サークルページの任意の場所およびサークルページに

50

表示されているキャラクタをマウスで直接選択することによってもキャラクタを移動させる位置を決定することができる。なお、アイコンフレームに設けられたリンクON/OFFボタン126を選択するとサークルページに表示されたウェブページのリンクを無効とすることができる。したがって、リンクがはられた画像などをマウスで選択した場合にもリンク先に繋がることなくキャラクタを移動させる位置情報として選択できる。移動場所を決定して追加ボタン164(図10参照)を選択すると、ステップS143において場所の移動であると判断し、ステップS145で移動情報をメモリ44に取り込む。「移動情報の取り込み」の処理は図22に示すフロー図の処理手順に従って処理される。「移動情報の取り込み」の処理では、何によってキャラクタの移動位置が指定されたかをステップS211およびステップS215で判断し、指定の方法に応じた処理を行う。

10

【0056】

プルダウンメニューからの画像名の選択およびウェブページ上の画像の直接クリックによってキャラクタを移動させる位置を指定した場合にはステップS211において、画像が選択されたと判断し、ステップS213でウェブページのURLおよびクリックされた画像のIDとを取得してメモリ44に取り込む。また、プルダウンメニューからのキャラクタ名の選択およびウェブページ上の他のキャラクタの直接クリックによってキャラクタを移動させる位置を指定した場合にはステップS215において、キャラクタが選択されたと判断し、指定されたキャラクタのキャラクタIDをステップS217で取得してメモリ44に取り込む。また、プルダウンメニューからのウェブページ上の位置(サークルページの右上など)の選択およびウェブページ上の任意の位置の直接クリックによってキャラクタを移動させる位置を指定した場合には、ステップS211およびステップS215においてNOと判断し、つまりウェブページ上の位置が指定されたと判断してステップS219において、指定されたウェブページ上の位置の座標を取得してメモリ44に取り込む。

20

【0057】

このようにしてメモリ44に取り込んだ移動情報に基づき、移動位置にキャラクタを移動させるためのスクリプトを図20のステップS139で生成し、生成したスクリプトをステップS141でスクリプトファイルに追加する。キャラクタをサークルページの右隅に移動させるとき、スクリプトは“<#move>rightside_of_the_circlepage”と記述され、かかるスクリプトが時系列構造でファイルに追加される。

30

【0058】

ユーザがプルダウンボタン158(図10参照)を選択すると振る舞いメニュー表示部154に、キャラクタの行うことができる振る舞いの一覧がプルダウンメニュー(図示せず)として表示される。振る舞いの一覧に表示される内容としては、「右を向く」、「左を指差す」、「驚きの仕草」、「微笑む」などが考えられる。これらの一覧から振る舞いを決定して追加ボタン162を選択すると、ステップS155において振る舞いであると判断する。そして決定された振る舞いの情報をステップS157においてメモリ44に取り込み、振る舞いの情報に基づいてステップS139でキャラクタに振る舞いを実行させるためのスクリプトを生成する。ステップS141では、生成されたスクリプトを時系列的にスクリプトファイルに追加する。キャラクタを微笑ませるとき、スクリプトは“<#play>smile”と記述される。

40

【0059】

さらに、ユーザがキーボード34などから台詞入力部172に台詞を入力して追加ボタン174を選択すると、ステップS135において台詞であると判断してステップS137で台詞入力部142に入力されたテキストデータをメモリ44に取り込む。そして取り込んだテキストデータに基づいてステップS139でキャラクタに台詞を発言させるためのスクリプトを生成し、生成したスクリプトをステップS141で時系列的にスクリプトファイルに追加する。キャラクタに“今日、こんなページを見つけたよ”という台詞をしゃべらせる場合、スクリプトは“<#speak>今日、こんなページを見つけたよ”と

50

記述される。

【0060】

このような一連の動作を選択することによって、サークルページに任意の画像を表示し、画像のそばに移動させたキャラクタに画像を指差させながら台詞を発言させるスクリプトファイルを生成することができる。そしてこの一連の動作の内容は履歴表示部176に表示される。なお、スクリプトが時系列構造で作成されることから、後にキャラクタエージェントの表現の内容を解析することは比較的容易である。つまり、<#____>の内容に基づいて、キャラクタエージェントの表現について種類毎に統計をとったり、所望の種類表現を検索することができる。

【0061】

これらの動作の他に、ユーザはURLで指定するウェブページをサークルページに表示することもできる。ユーザはキーボード34などを用いてURL入力部166にURLを入力して追加ボタン168を選択する。するとステップS147においてURLであると判断する。そしてステップS149においてURLのテキストデータをメモリ44に取り込む。ステップS139で、URLのテキストデータに基づいてURLによって指定されるウェブページを表示させるためのスクリプトを生成し、生成したスクリプトをステップS141でスクリプトファイルに追加する。このとき、スクリプトは、たとえば“<#open>http//www.mic.atr.co.jp”と記述される。

【0062】

このようなスクリプトを生成することによってURLで指定される任意のウェブページをサークルページ上に、もしくはサークルページとして表示できる。こうして表示されるウェブページはサーバ10でプロキシ(Proxy)処理を施されることによってサークルページと同様の機能を備えることができる。なお、プロキシ処理の詳細については後述する。

【0063】

このような操作を新規話題編集用ウィンドに対して行うことによりユーザは新規話題の作成および最初のコメントを作成する。そしてユーザが完成ボタン178(図10参照)を操作するとステップS159において完成であると判断し、ステップS161においてLv-SWに設定されている値が「0」であるかどうか判断する。Lv-SWに「0」が設定されている、つまり新規話題の編集であると判断すると、ユーザが予めタイトル入力欄152に入力しておいた話題のタイトルのテキストデータをステップS163においてメモリ44に取り込む。そして取り込んだタイトルのテキストデータをステップS165でスクリプトファイルに追加する。さらに、ステップS167においてLv-SWに設定されている値およびキャラクタのキャラクタIDをスクリプトファイルに追加してスクリプトファイルをクローズする。ステップS169では、Lv-SWの値に「0」を設定してクリアするとともに、スクリプトファイルをサーバ10に送信する。なお、新規話題編集の場合にはステップS169の前後においてLv-SWに設定されている値は変わらない。

【0064】

サーバ10では図21に示すようにスクリプトファイルを受信し、ステップS181、ステップS199およびステップS205においてスクリプトファイルに含まれているLv-SWの値が「0」、「1」および「2」のいずれであるかを判断する。新規話題の編集ではステップS181においてLv-SWに設定されている値が「0」であると判断し、ステップS183において新規話題のスレッドをHD22に設けられたスレッドデータベース80(図7参照)内に生成する。具体的には、当該サークル(たとえば図8におけるサークル1)のフォルダに新規話題(たとえば話題1)の話題フォルダ90を生成し、話題フォルダ90の内に登場キャラクタ情報92を生成する。さらに話題フォルダ90の内にステップS185で場面フォルダ94を生成し、場面フォルダ94内にクライアント30から受信したスクリプトファイルをステップS187でコメントとして格納する。なお、この場合は新規話題であるから場面1の場面フォルダ94が生成される。

10

20

30

40

50

【 0 0 6 5 】

そしてステップ S 1 8 9 においてスクリプトファイルに記録されているキャラクタ ID を登場キャラクタ情報 9 2 に記録されているキャラクタ ID と比較し、新規キャラクタであるかどうか判断する。新規キャラクタであるときはステップ S 1 9 1 において当該キャラクタのキャラクタ ID を登場キャラクタ情報 9 2 に追加する。

【 0 0 6 6 】

スレッドデータベース 8 2 の更新（話題フォルダ 9 0 への場面フォルダ 9 4 の追加）が終了した後、更新された話題フォルダ 9 0 に記録されている情報に基づいてサークルデータベース 8 8 に記録されている当該サークル（たとえばサークル 1 ）のスレッドページをステップ S 1 9 3 において更新する。そして、更新したスレッドページをステップ S 1 9 5 においてクライアント 3 0 に送信する。クライアント 3 0 では更新されたスレッドページ（図 4 参照）を受信し、ステップ S 1 9 7 においてディスプレイ 4 0 に表示する。更新されたスレッドページには、新たな話題ボタン 1 2 0 およびコメント 1 が追加される。こうして「スクリプトファイル生成」の処理（図 2 0 参照）が終了するとともに、「新規話題作成」の処理（図 1 7 のステップ S 8 9 参照）が終了する。

【 0 0 6 7 】

新規話題（最初のコメントを含む）が作成されると、当該ユーザおよびサークルに入場した他のユーザは、話題に次のコメント（たとえばコメント 1 に対するコメント 2 ）を追加し、もしくはコメントに対するキャラクタのリアクションを追加することができる。しかし、リアクションは当該場面のすべてのコメントおよびリアクションが再現された後、つまりユーザがその場面のすべてを見終えた後にのみ追加することができる。また、新たなコメントは当該話題のすべての場面を見終えた後にのみ追加することができる。そのため、既存の話題を再現する手順から先に説明する。

【 0 0 6 8 】

ユーザがスレッドページの話題ボタン 1 2 0 （たとえば話題 1 ）を選択すると、クライアント 3 0 では図 1 7 のステップ S 8 5 において操作ありと判断し、さらにステップ S 9 5 で話題の再現が要求されたと判断する。そして、ステップ S 9 7 において「話題再現」の処理を実行する。

【 0 0 6 9 】

「話題再現」の処理は、図 2 3 から図 2 8 のフロー図に示す手順に従って実行される。まず、クライアント 3 0 ではステップ S 2 2 1 において、選択された話題を選択された話題ボタン 1 2 0 に基づいて特定し、特定した話題の最初の場面（場面 1 ）の再現要求をステップ S 2 2 3 でサーバ 1 0 に送信する。サーバ 1 0 ではステップ S 2 2 5 で再現要求を受け取り要求された場面を再現するための場面 HTML ファイルの雛型を雛型データベース 8 6 （図 7 参照）からメモリ 2 4 に取得し、オープンする。要求された場面を再現する HTML ファイルは、HTML ファイルの雛型に場面フォルダ 9 4 （図 8 参照）に格納されているスクリプトファイル（コメントおよびリアクション）の内容を埋め込むことによって生成される。

【 0 0 7 0 】

ステップ S 2 2 7 では、再現が要求されたのが当該話題の最初の場面であるかどうかを再現要求に基づいて判断する。最初の場面であると判断すると、ステップ S 2 2 9 において話題フォルダ 9 0 内の登場キャラクタ情報 9 2 を参照し、当該話題に登場するすべてのキャラクタを特定する。ステップ S 2 3 1 では、特定したすべてのキャラクタのデータをキャラクタ ID に基づいてキャラクタデータベース 7 4 のデフォルトキャラクタデータベース 7 6 もしくはオリジナルキャラクタデータベース 7 8 から取得する。そしてステップ S 2 3 3 でこれらのキャラクタデータをバッファ 1 6 に格納する。

【 0 0 7 1 】

ステップ S 2 3 5 では、再現が要求された場面の場面フォルダ 9 4 に含まれているスクリプトファイルをすべてスレッドデータベース 2 2 5 から取得してメモリ 2 4 に格納する。まず、ステップ S 2 3 7 でコメントのスクリプトファイルに記録されている内容を H T

10

20

30

40

50

M L ファイルの所定の位置に埋め込み、ステップ S 2 3 9 でさらに他のスクリプトファイル（リアクションのスクリプトファイル）が存在するかどうか判断する。スクリプトファイルが存在するときにはステップ S 2 4 1 でリアクションのスクリプトファイルの内容を H T M L ファイルの雛型に埋め込む。そして、ステップ S 2 3 9 でスクリプトファイルが存在しないと判断するまでステップ S 2 3 9 およびステップ S 2 4 1 を繰り返してすべてのリアクションスクリプトファイルの内容を H T M L ファイルの雛型に埋め込み、ステップ S 2 4 3 で場面 H T M L ファイルをクローズする。こうして生成した場面 H T M L ファイルをステップ S 2 4 5 においてバッファ 1 6 に格納し、ステップ S 2 4 7 でバッファ 1 6 に格納されている場面 H T M L ファイルを（場面 1 の場合には、キャラクタデータとともに）クライアント 3 0 に送信する。

10

【 0 0 7 2 】

クライアント 3 0 では、図 2 4 のステップ S 2 5 1 において場面 H T M L ファイル（キャラクタデータを含む場合もある）を受信し、ステップ S 2 5 3 でキャラクタデータが存在するかどうか判断する。そしてキャラクタデータが存在するときには図 5 に示すようにすべてのキャラクタをサークルページ上に表示する。図 5 の例では 3 人のユーザ（キャラクタ 1 ~ 3 ）が話題に参加している。

【 0 0 7 3 】

ステップ S 2 5 7 において、場面 H T M L ファイルに埋め込まれているスクリプトの処理がすべて終了したかどうか判断し、終了していないときにはステップ S 2 5 9 , ステップ S 2 7 1 , ステップ S 2 8 1 （図 2 5 ）, ステップ S 3 0 1 およびステップ S 3 0 7 においてスクリプトの内容を判断する。

20

【 0 0 7 4 】

スクリプトが U R L に基づくウェブページの表示を示しているときには、ステップ S 2 5 9 において U R L であると判断される。そしてステップ S 2 6 1 において、U R L で指定されるウェブページのプロキシ処理要求を U R L とともにサーバ 1 0 に送信する。

【 0 0 7 5 】

サーバ 1 0 ではステップ S 2 6 3 においてプロキシ処理要求を受信し、ステップ S 2 6 5 で「プロキシ処理」を実行する。この「プロキシ処理」が施されることによって、U R L で指定されるウェブページがサークルのウェブページとしての機能を備えることができる。「プロキシ処理」は、図 2 8 のフロー図に示す手順に沿って実行される。まず、ステップ S 3 4 1 において U R L で指定されるサーバにアクセスする。そしてステップ S 3 4 3 で、U R L で指定されるウェブページを形成する H T M L データを 1 行分だけメモリ 2 4 に取得し、ステップ S 3 4 5 においてメモリ 2 4 に取得したデータを加工する「データの加工処理（プロキシ処理）」を行う。

30

【 0 0 7 6 】

「データの加工処理」では、メモリ 2 4 に取得した 1 行分のデータにイメージタグが存在する場合にはその画像（イメージ）に I D を割り振るとともに、「スクリプトの生成」の処理において画像をクリックしたときにキャラクタを移動させる位置情報として当該画像の位置を取り込み、プルダウンボタン 1 6 0 （図 1 0 参照）もしくはプルダウンボタン 2 0 0 （図 1 1 参照）が選択されたときにはプルダウンメニュー表示部 1 5 6 もしくはプルダウンメニュー表示部 1 9 8 に表示される選択メニューに当該画像を表示するためのデータを付加する。なお、このデータは後述するスクリプトプログラムによって実行される。

40

【 0 0 7 7 】

1 行分のデータにリンクタグが存在する場合には、リンクタブに設定されている U R L を、サーバ 1 0 のプロキシ経由で U R L が指定するウェブページにアクセスするように書き換える。具体的には、U R L が指定するアクセス先をサーバ 1 0 に変更し、この U R L に元の U R L が指定するウェブページからデータを取得してプロキシ処理を行うというオプションを付加する。

【 0 0 7 8 】

50

また、1行分のデータにフレームタグが存在する場合には、リンクタグの場合と同様にフレームタグに設定されているURLをサーバ10のプロキシ経由でアクセスするように書き換える。

【0079】

加工が終了した一行分のデータはステップS347でバッファ16に格納される。URLで指定されるウェブページを形成するHTMLのすべての行についてデータの加工（プロキシ処理）が行われるとステップS349でデータが終了したと判断し、ステップS351でスクリプトプログラムをバッファ16に格納する。バッファ16に格納することによってURLで指定されるウェブページのHTMLにスクリプトプログラムが挿入される。プロキシ処理によって加工されたデータはこのスクリプトプログラムによって実行される。スクリプトプログラムの働きにより「スクリプトの生成」の処理（図20参照）においてウェブページ上の画像をクリックされると当該画像の位置がキャラクタを移動させる位置情報として取り込まれるとともに、当該画像の位置までキャラクタが移動される。また、「話題再現」（スクリプトの再生）の処理（図23参照）において再生されるスクリプトの内容が画像の位置への移動や振る舞いである場合には、スクリプトプログラムがスクリプトを処理してキャラクタを画像の位置に移動させたり振る舞いを実行させたりする。そして図24のステップS267では、バッファ16に格納されているプロキシ処理済のウェブページをクライアント30に送信する。

10

【0080】

クライアント30はステップS269においてウェブページをプロキシサーバ経由で取得してサークルページ上に表示する。こうして表示されたウェブページは図5に示すようにサークルページの背景（つまりサークルページ自体）とすることもできるが、ウェブページの内容（たとえば画像302）を表示するための別ウィンド300を展開するようにすることもできる。

20

【0081】

また、スクリプトがユーザによってアップロードされた画像などのローカルファイルの表示を示しているときには、ステップS271においてファイルであると判断される。そして、ステップS273において表示位置を特定し、ステップS275で図5に示すように画像302を含む別ウィンド300をサークルページ上に表示する。

【0082】

スクリプトがキャラクタの移動を示しているときには、図25のステップS281において移動であると判断され、ステップS283において移動の対象となるキャラクタ（たとえばキャラクタ1）を特定する。そして、スクリプトに定義されている移動先の指定方法に応じた処理を行う。

30

【0083】

移動先がウェブページ上の画像の選択によって指定されているときにはステップS285においてウェブページ上の画像の選択であると判断し、ステップS287でスクリプトに基づいてウェブページのURLおよび移動先である画像IDを特定する。そして、ステップS289でキャラクタ1を画像IDによって特定される画像の位置に移動させる。このとき、キャラクタ1は図6に示すようにMicrosoft Agentなどのプログラムによる処理でアニメーションしながら移動するようにするとよい。このようにしてキャラクタ1はたとえば図6に示すように画像302の右側へと移動する。また、移動先が他のキャラクタの選択によって指定されているときにはステップS291においてキャラクタの選択であると判断し、ステップS293においてスクリプトに基づいて移動先であるキャラクタのキャラクタIDを特定する。さらに特定したキャラクタのサークルページにおける座標をステップS295で特定する。そして、ステップS289でキャラクタ1を座標で特定される移動先に移動させる。ステップS285およびステップS291での判断がNOであると移動先は座標によって指定されていると判断し、ステップS297でスクリプトに基づいて移動先の座標を特定する。そして、ステップS289でキャラクタ1を特定した座標位置まで移動させる。

40

50

【 0 0 8 4 】

つぎに、スクリプトがキャラクタの振る舞いを示すときには、図 2 6 のステップ S 3 0 7 において振る舞いであると判断され、ステップ S 3 0 9 において振る舞いを行わせる対象となるキャラクタ（たとえばキャラクタ 1）を特定する。そしてステップ S 3 1 1 においてキャラクタ 1 をアニメーションさせるなどしてスクリプトに定義された振る舞い、たとえば左側を指差すなどの振る舞いを行わせる。

【 0 0 8 5 】

さらに、スクリプトが台詞の発言を示すときには、ステップ S 3 0 1 において台詞であると判断され、ステップ S 3 0 3 で台詞の発言の対象となるキャラクタ（たとえばキャラクタ 1）を特定する。そしてステップ S 3 0 5 で、特定したキャラクタの近傍で図 6 に示すように他のキャラクタやウィンド 3 0 0 などに重ならない適切な位置に吹き出し 3 0 4 を表示する。なお、吹き出し 3 0 4 の中には、スクリプトに定義されている台詞、たとえば「こんな写真を見つけたよ。どう？」などのテキストを表示する。

【 0 0 8 6 】

以上説明したように、場面のコメントではサークルページ上にウェブページおよび画像を表示し、キャラクタに移動、振る舞いおよび台詞の発言をさせることができる。以上のウェブページおよび画像の表示ならびにキャラクタ 1 の動作を場面 1 におけるコメント 1 とすると、図 8 の場面フォルダ 9 4 からわかるように場面 1 の場面 H T M L ファイルにはコメント 1 に対するリアクションのスクリプトがさらに存在する場合がある。

【 0 0 8 7 】

リアクションのスクリプトが存在するときにはステップ S 3 0 7 において振る舞いであると判断され、スクリプトに従って、たとえばキャラクタ 2 がアニメーションによる振る舞いが再現される。図 6 の例では、キャラクタ 1 によるコメントに対してキャラクタ 2 が驚きのリアクションをとっている。リアクションのスクリプト生成の説明で詳しく述べるが、リアクションのスクリプトにはキャラクタの振る舞いに関する情報のみが記録されている。

【 0 0 8 8 】

こうして場面 H T M L ファイルに含まれているすべてのスクリプトの再現が終了すると図 2 4 のステップ S 2 5 7 においてスクリプト終了と判断し、図 2 7 のステップ S 3 2 1 でユーザによるボタン操作があるかどうか判断する。なお、場面 H T M L ファイルに含まれるすべてのスクリプトの再現が終了すると、リアクション作成ボタン 1 1 6 および次の場面ボタン 1 1 8 が有効となる。

【 0 0 8 9 】

ユーザが次の場面ボタン 1 1 8 を選択するとステップ S 3 2 3 で次の場面が要求されたと判断し、当該話題の次の場面についての再現要求をステップ S 3 2 5 でサーバ 1 0 に送信する。

【 0 0 9 0 】

サーバ 1 0 では再現要求を受信し、図 2 3 のステップ S 2 2 5 で場面 H T M L ファイルの雛型をオープンして場面 H T M L ファイルの生成を開始する。前述したのと同じ手順によってたとえば場面 2 の場面 H T M L ファイルが生成されてクライアント 3 0 に送信される。そしてクライアント 3 0 で場面 H T M L ファイルに基づいて場面 2 が再現される。場面 2 ではたとえば、キャラクタ 3 がキャラクタ 1 のコメントおよびキャラクタ 2 のリアクションを受けて、キャラクタ 2 の下方に移動し、吹き出し 3 0 6 を用いて台詞「きれいな朝焼けですね」などの発言を行う。

【 0 0 9 1 】

1 つの場面 H T M L ファイルに含まれるすべてのスクリプトが再現されると、リアクション作成ボタン 1 1 6 が有効となり当該場面にリアクションを追加することができるようになる。ユーザがリアクション作成ボタン 1 1 6 を選択すると図 2 7 のステップ S 3 2 7 においてリアクション編集であると判断し、ステップ S 3 2 9 で図 1 2 に示すようなリアクション編集用ウィンド 2 2 0 をディスプレイ 4 0 に表示する。そしてステップ S 3 3 1

10

20

30

40

50

においてLv - SWに「2」を設定してステップS333で「スクリプトファイル生成」の処理を実行する。

【0092】

前述したように「スクリプトファイル生成」の処理は図20から図22のフロー図に示す手順で実行される。図12のリアクション編集用ウィンド220からもわかるように、リアクション編集ではキャラクタによる振る舞いのみを編集することができる。

【0093】

まず、図20のステップS131において当該リアクション用のスクリプトファイルをオープンする。そしてステップS133においてLv - SWに設定されている値が「2」であるかどうか判断する、リアクション編集ではLv - SWには値「2」が設定されているのでステップS155に進む。

10

【0094】

ユーザがリアクション編集ウィンド220のプルダウンボタン224を選択すると、振る舞いメニュー表示部222にキャラクタの行うことができる振る舞いの一覧が表示される。一覧メニューから振る舞いを決定して追加ボタン226を選択すると、ステップS155において振る舞いが選択されたと判断され、決定された振る舞いの情報をステップS157においてメモリ44に取り込む。そしてステップS139で振る舞いの情報に基づいてキャラクタに振る舞いを実行させるためのスクリプトを生成し、生成されたスクリプトをステップS141でスクリプトファイルに追加してステップS133に戻る。

【0095】

20

こうしてユーザが完成ボタン230を選択するまでユーザが選択した振る舞いを表すスクリプトがスクリプトファイルに追加される。ユーザが完成ボタン230を選択するとステップS159においてスクリプトファイルが完成したと判断し、さらにステップS161でLv - SWの値が「0」であるかどうか判断する。Lv - SWに設定されている値は「2」であるからタイトルの取り込み処理をスキップし、ステップS167でLv - SWに設定されている値およびキャラクタのキャラクタIDをスクリプトファイルに追加してスクリプトファイルをクローズする。そして、ステップS169でLv - SWに値「0」を設定してクリアするとともにスクリプトファイルをサーバ10に送信する。

【0096】

サーバ10ではスクリプトファイルを受信し、ステップS205においてスクリプトファイルに含まれるLv - SWの値が「2」である、つまりアクションのスクリプトファイルを受信したと判断する。そして、ステップS207においてスレッドデータベース82（図7参照）に含まれている当該話題フォルダ90内の当該場面フォルダ94（たとえば場面2の場面フォルダ）にスクリプトファイルを格納する。つぎに、ステップS189において登場キャラクタ情報92に記録されているキャラクタIDをスクリプトファイルに記録されているキャラクタIDと比較し、新規のキャラクタであるかどうか判断する。新規キャラクタであるときにはステップS191において当該キャラクタIDを登場キャラクタ情報92に追加する。スレッドデータベース82の更新が終了した後ステップS193で、更新された話題フォルダ90の内容に基づいてサークルデータベース88に記録されている当該サークルのスレッドページを更新する。そして更新したスレッドページをステップS195でクライアント30に送信する。

30

40

【0097】

クライアント30ではスレッドページ（図4参照）を受信してステップS197でディスプレイ40に表示する。こうして図27のステップS333におけるリアクションの「スクリプトファイル生成」の処理が終了する。そしてステップS335において、リアクションを追加した当該場面（場面2）の再現要求をサーバ10に送信する。

【0098】

サーバ10では再現要求を受信し、上述した図23のステップS225からステップS247の処理を経てリアクションが追加された場面の場面HTMLファイルがクライアント30に送信され、クライアント30では受信した場面HTMLファイルに基づいて当該

50

場面がサークルページに再現される。こうしてサークルの場面が順次再現されるが、ユーザが終了ボタン308を選択するとステップS307において終了と判断し、図17のステップS97における「話題再現」の処理が終了する。

【0099】

当該話題（たとえば話題2）に含まれるすべての場面の再現が終了すると、上述したようにコメント作成ボタン114が有効となり、話題に場面（コメント）を追加することが可能になる。ユーザがコメント作成ボタン114を選択すると図17のステップS91においてコメントの作成であると判断し、ステップS93において「コメント作成」の処理を行う。「コメント作成」の処理は図19に示すフロー図の手順に従って実行される。まず、ステップS111において図11に示すようなコメント編集用ウィンド190をディスプレイ40に表示し、Lv-SWに「1」の値を設定する。そしてステップS115において「スクリプトファイル生成」の処理を実行する。前述したように「スクリプトファイル生成」の処理は図20から図22のフロー図に示す手順に従って実行される。

10

【0100】

「コメント作成」の処理における「スクリプトファイル生成」の処理は、「新規話題作成」の処理における「スクリプトファイル生成」の処理とほぼ同じであり、クライアント30における処理（図20参照）では生成したスクリプトファイルに話題のタイトルが追加されない点のみが異なる。また、この「コメント作成」においてはサーバ10における処理（図22参照）では新規に話題フォルダ90（新規スレッド）が生成されず、既存の話題フォルダ90に場面フォルダ94を生成してコメントのスクリプトファイルが格納される点のみが異なる。したがって「コメント作成」の処理の詳細な説明を省略する。なお、フロー図には特に示さなかったがアイコンフレームに設けられた、戻るボタン124を選択すると、話題を選択するページやサークルを選択するページに戻るようにするとよい。

20

【0101】

以上に説明したように、この実施例によれば、テキストデータのみという単次元の手段で情報交換を行う従来のBBSと異なり、サークルのウェブページ上に情報源である画像ファイルや他のウェブページなどを実際に表示させるとともに、ユーザはユーザの代理であるキャラクタを使ってサークルのウェブページ上で移動し、動作し、台詞を発言することができる。したがって、コミュニティ環境提供システムのサークルに参加するユーザは、多次元の表現手段によってより質が高くより正確な情報交換をスムーズに行うことができる。

30

【0102】

また、キャラクタエージェントの表現を制御するためのスクリプトは、メニューから所望の移動先や振る舞いを選択したり、台詞を入力することによって作成されるため、キャラクタエージェントの表現を容易に制御することができる。

【0103】

なお、この実施例は上述の例に限るものではなく、種々に変更して実施してもよい。たとえば、サークルでの表現手段はキャラクタによる動作、ウェブページの表示および画像や動画などの映像データとしたが、これらに限らず音楽を再生するなどウェブページ上で実現可能であればどのような表現手段であってもよい。

40

【0104】

また、この実施例ではサークル，話題，場面という概念を用いたが、必ずしもこのような概念を用いる必要はない。さらに、場面は1つのコメントもしくは1つのコメントと少なくとも1つのリアクションを含む構成としたが、場面の構成もこれに限るものではない。

【0105】

さらに、場面に含まれるすべての表現が再現された後にリアクションを追加可能とし、話題に含まれるすべての場面が再現された後に場面の追加を可能としたが、リアクションおよび場面の追加条件はこのようなものに限らず、よりスムーズな表現および情報交換を

50

実現し、より快適なコミュニティ環境を実現できるように適宜変更してもよい。

【図面の簡単な説明】

【0106】

【図1】この発明の一実施例の全体構成を示す図解図である。

【図2】サーバの構成の一例を示すブロック図である。

【図3】クライアントの構成の一例を示すブロック図である。

【図4】サークルの入り口ページの一例を示す図解図である。

【図5】再現された場面の一例を示す図解図である。

【図6】再現された場面の他の一例を示す図解図である。

【図7】サーバに設けられたH Dのデータ構成の一例を示す図解図である。

10

【図8】H Dに構築されたスレッドデータベースのデータ構成の一例を示す図解図である。

【図9】コミュニティ環境提供システムへのログインページの一例を示す図解図である。

【図10】新規話題を編集するためのエディタの一例を示す図解図である。

【図11】コメントを編集するためのエディタの一例を示す図解図である。

【図12】リアクションを編集するためのエディタの一例を示す図解図である。

【図13】図1の実施例における処理手順の一部を示すフロー図である。

【図14】図1の実施例における処理手順の他の一部を示すフロー図である。

【図15】図1の実施例における処理手順の他の一部を示すフロー図である。

【図16】図1の実施例における処理手順の他の一部を示すフロー図である。

20

【図17】図1の実施例における処理手順の他の一部を示すフロー図である。

【図18】図1の実施例における処理手順の他の一部を示すフロー図である。

【図19】図1の実施例における処理手順の他の一部を示すフロー図である。

【図20】図1の実施例における処理手順の他の一部を示すフロー図である。

【図21】図1の実施例における処理手順の他の一部を示すフロー図である。

【図22】図1の実施例における処理手順の他の一部を示すフロー図である。

【図23】図1の実施例における処理手順の他の一部を示すフロー図である。

【図24】図1の実施例における処理手順の他の一部を示すフロー図である。

【図25】図1の実施例における処理手順の他の一部を示すフロー図である。

【図26】図1の実施例における処理手順の他の一部を示すフロー図である。

30

【図27】図1の実施例における処理手順の他の一部を示すフロー図である。

【図28】図1の実施例における処理手順の他の一部を示すフロー図である。

【符号の説明】

【0107】

10 ...サーバ

12 ...CPU

22 ...H D (Hard Disc)

30 ...クライアント

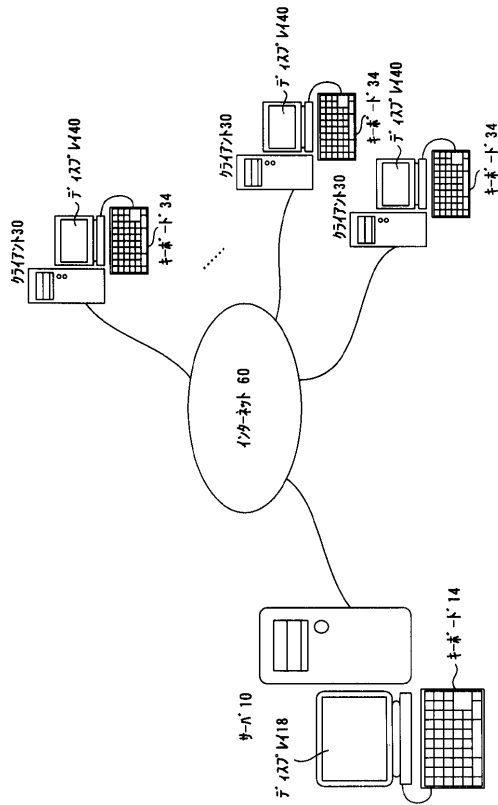
32 ...CPU

42 ...H D (Hard Disc)

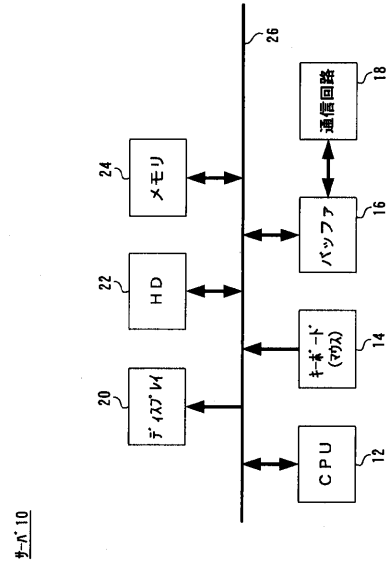
40

60 ...インターネット

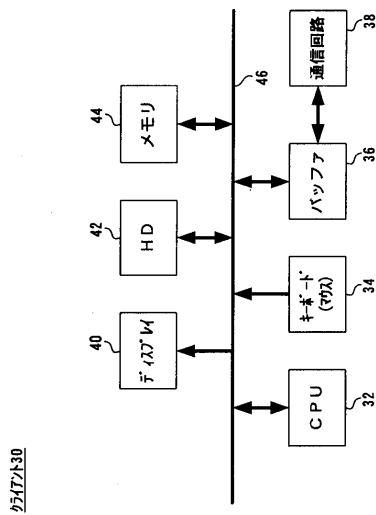
【図 1】



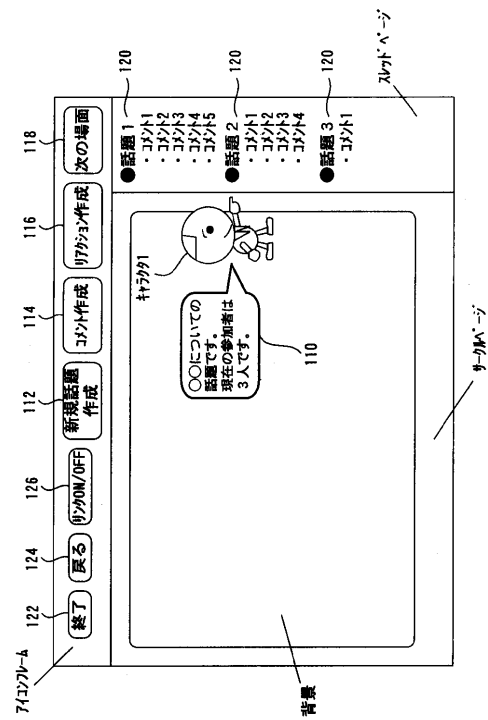
【図 2】



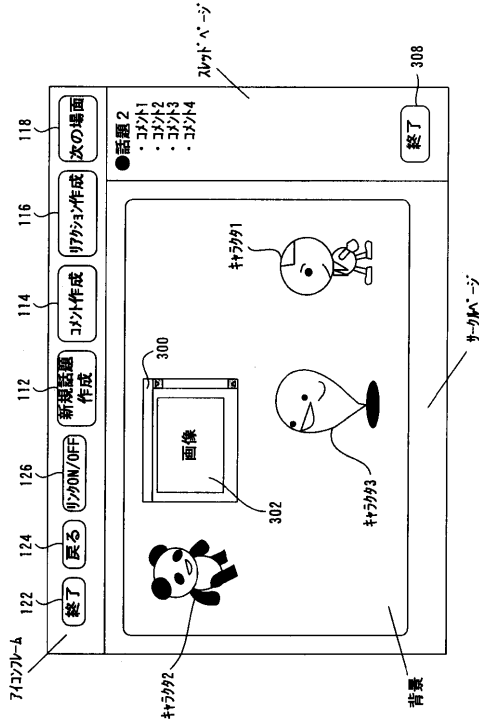
【図 3】



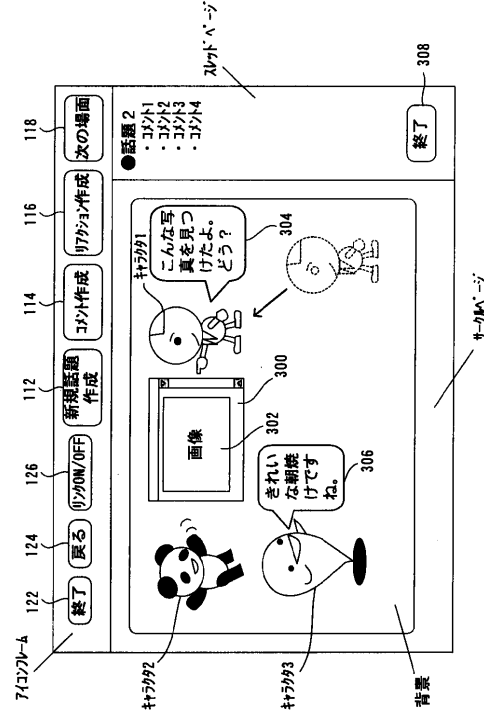
【図 4】



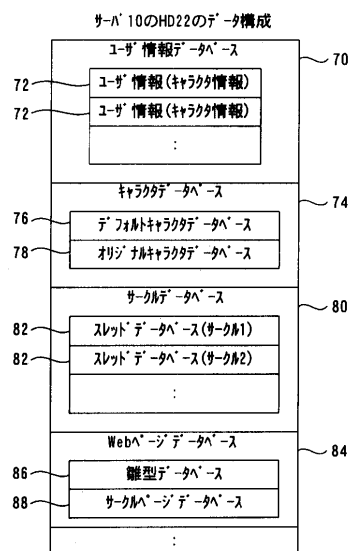
【図 5】



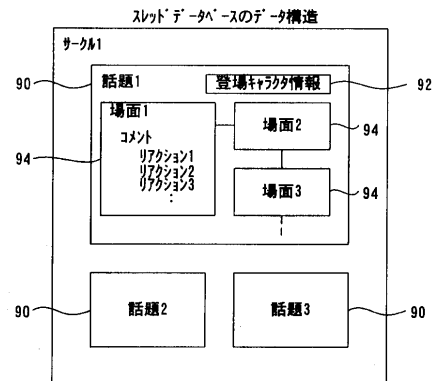
【図 6】



【図 7】



【図 8】



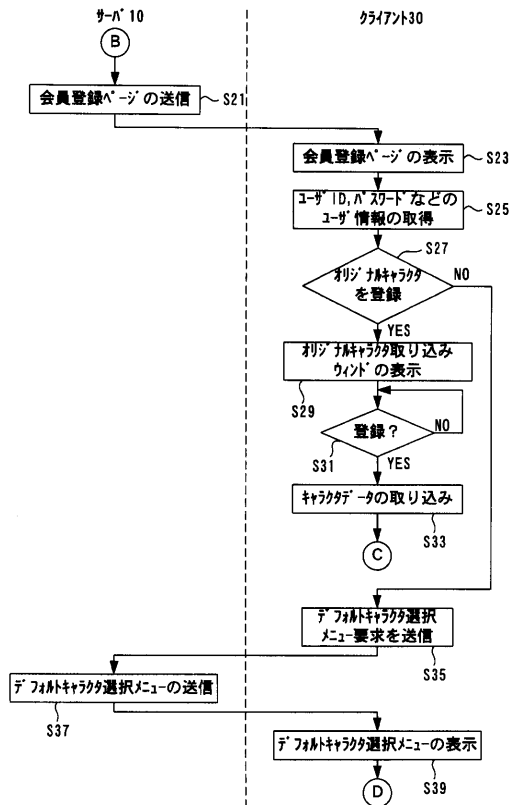
【 ㊦ 1 0 】

【 図 1 3 】

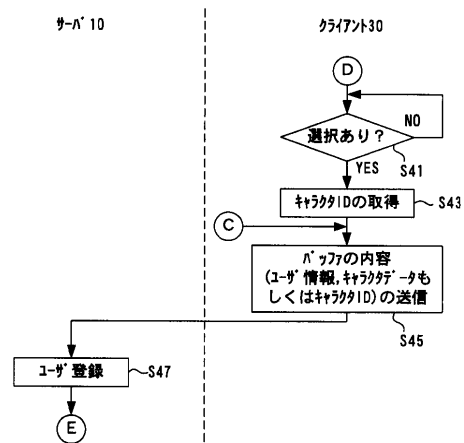
```

graph TD
    Start([開始]) --> S1[S1: クライアント30からの接続要求の送信]
    S1 --> S3[S3: クライアントとの接続を確立]
    S3 --> A((A))
    A --> S5[S5: ログインページの送信]
    S5 --> S7[S7: ログインページの表示]
    S7 --> S9{S9: 会員?}
    S9 -- YES --> S13[S13: ユーザ名、パスワードの取り込み]
    S9 -- NO --> S11[S11: 会員登録ページの要求の送信]
    S11 --> B((B))
    B --> S13
    S13 --> S15[S15: ユーザ名、パスワードを送信]
    S15 --> S19{S19: 認証?}
    S19 -- YES --> E((E))
    S19 -- NO --> A
  
```

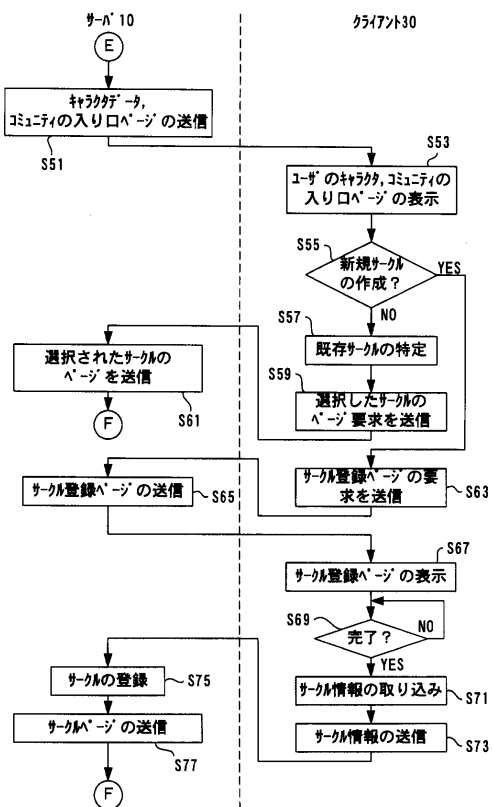
【図 14】



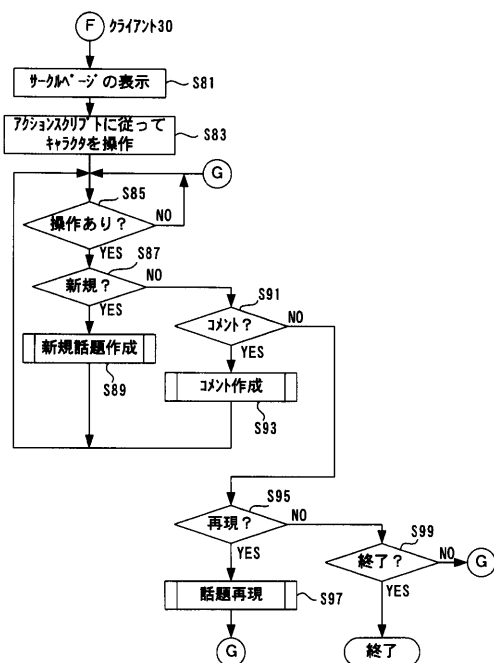
【図 15】



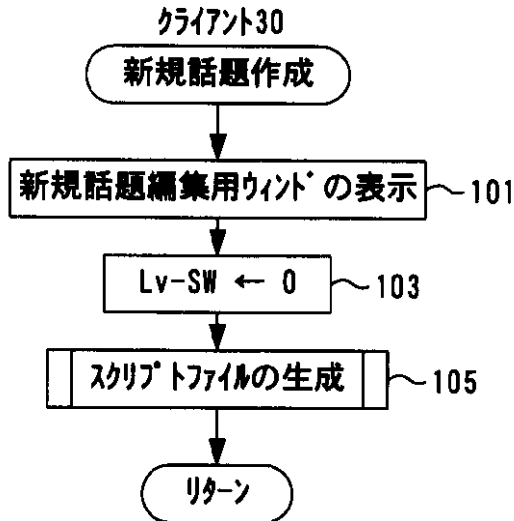
【図 16】



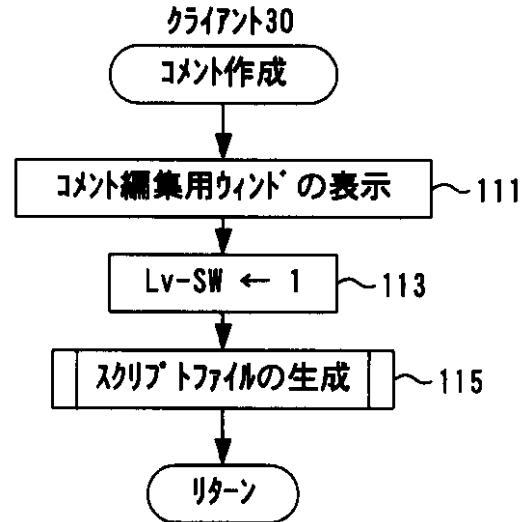
【図 17】



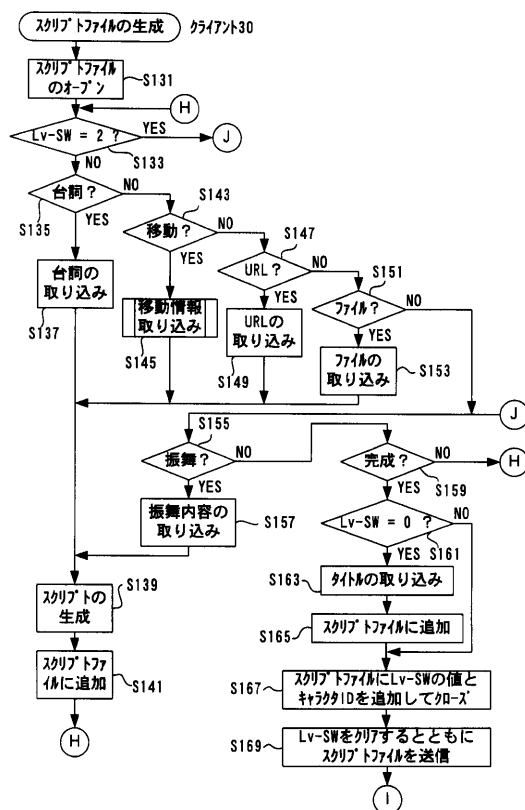
【図18】



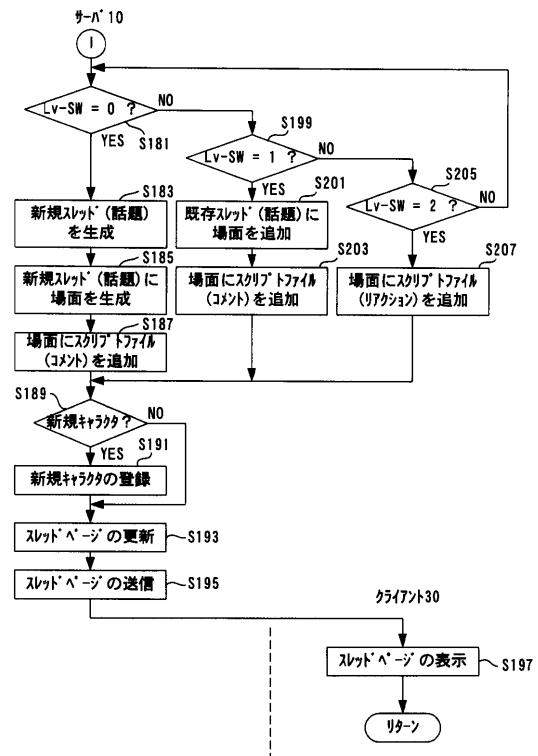
【図19】



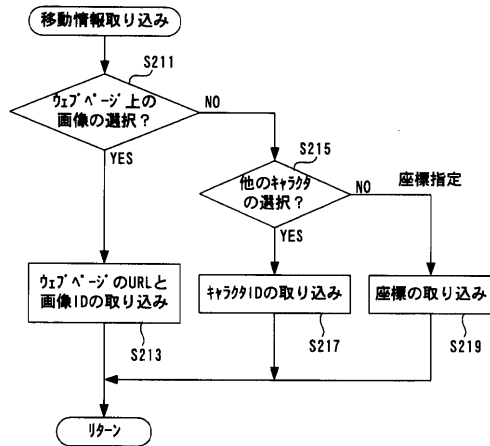
【図20】



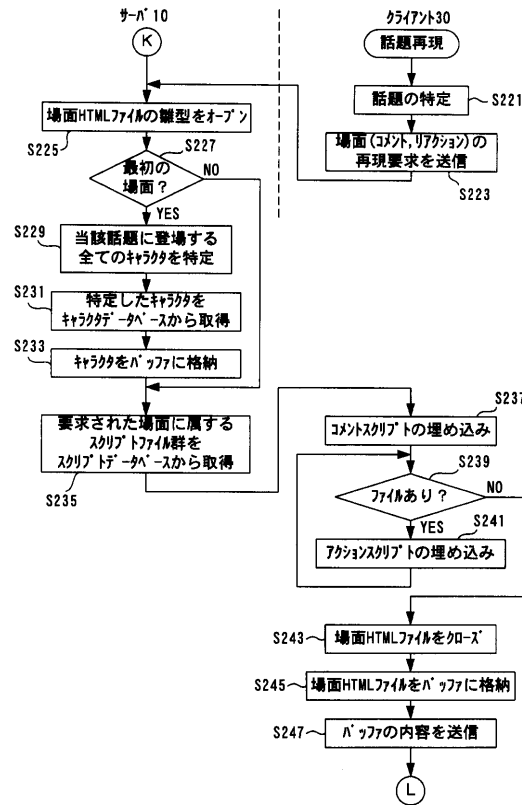
【図21】



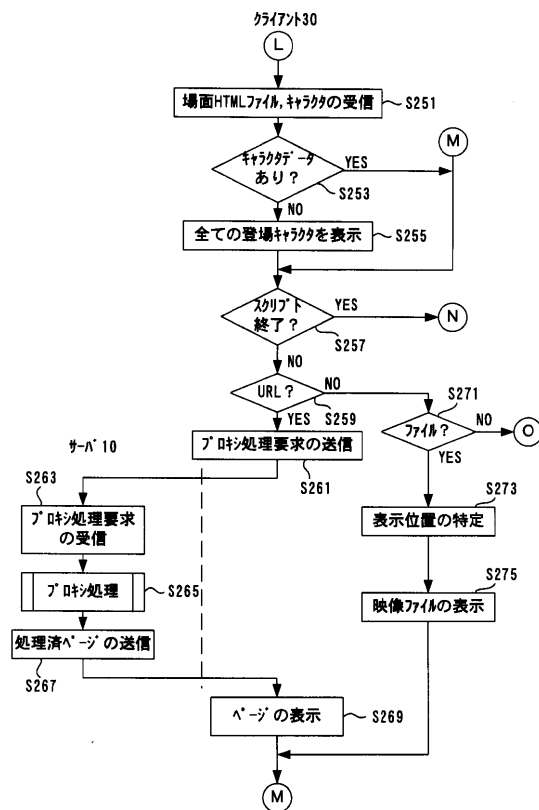
【図 22】



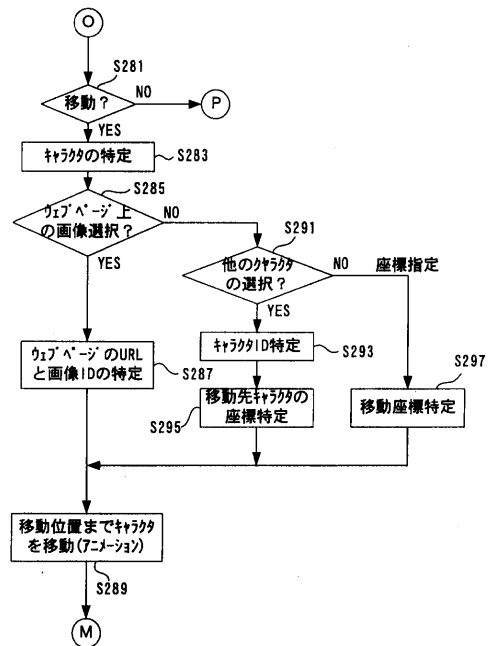
【図 23】



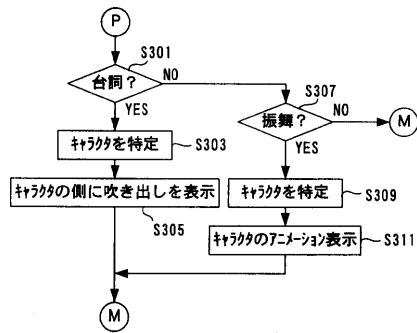
【図 24】



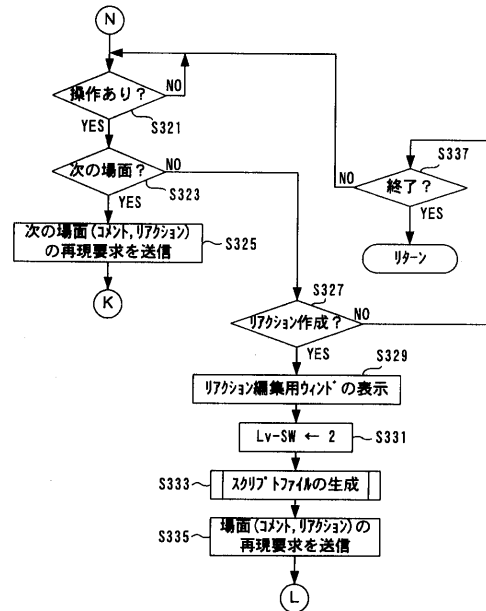
【図 25】



【図 26】



【図 27】



【図 28】

