

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6855348号  
(P6855348)

(45) 発行日 令和3年4月7日(2021.4.7)

(24) 登録日 令和3年3月19日(2021.3.19)

(51) Int.Cl.

F 1

G06F 13/00 (2006.01)

G06F 13/00 530A

A63F 13/358 (2014.01)

A63F 13/358

G06F 16/182 (2019.01)

G06F 16/182

請求項の数 7 (全 14 頁)

(21) 出願番号 特願2017-148128 (P2017-148128)  
 (22) 出願日 平成29年7月31日 (2017.7.31)  
 (65) 公開番号 特開2019-28778 (P2019-28778A)  
 (43) 公開日 平成31年2月21日 (2019.2.21)  
 審査請求日 令和1年9月10日 (2019.9.10)

前置審査

(73) 特許権者 310021766  
 株式会社ソニー・インタラクティブエンタテインメント  
 東京都港区港南1丁目7番1号  
 (74) 代理人 100105924  
 弁理士 森下 賢樹  
 (74) 代理人 100109047  
 弁理士 村田 雄祐  
 (74) 代理人 100109081  
 弁理士 三木 友由  
 (74) 代理人 100134256  
 弁理士 青木 武司

最終頁に続く

(54) 【発明の名称】情報処理装置およびダウンロード処理方法

(57) 【特許請求の範囲】

【請求項 1】

情報処理装置であって、

当該情報処理装置においてユーザにより選択されている使用言語を特定する使用言語情報保持する言語情報保持部と、

アプリケーションソフトウェアを記憶装置にダウンロードするダウンロード処理部と、を備え、アプリケーションソフトウェアは、アプリケーション本体と、複数言語の言語依存ファイルを含んでおり、

前記ダウンロード処理部は、アプリケーションソフトウェアのダウンロード開始前に、アプリケーション本体と、前記言語情報保持部に保持されている使用言語情報により特定される使用言語の言語依存ファイルのデータサイズをサーバに問い合わせ、使用言語情報により特定される使用言語ではない言語の言語依存ファイルのデータ領域については記憶装置に確保せず、アプリケーション本体と、使用言語情報により特定される使用言語の言語依存ファイルのデータ領域を記憶装置に確保してから、ダウンロードを実行するものであって、

アプリケーションで使用する言語に変更が生じた場合に、前記ダウンロード処理部は、変更後の使用言語の言語依存ファイルのデータサイズをサーバに問い合わせ、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認することを特徴とする情報処理装置。

【請求項 2】

10

前記ダウンロード処理部は、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できない場合に、ダウンロードエラーに関する通知を表示する、ことを特徴とする請求項1に記載の情報処理装置。

#### 【請求項3】

前記ダウンロード処理部は、連続領域の所定のサイズを最小単位として、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認する、ことを特徴とする請求項1または2に記載の情報処理装置。

#### 【請求項4】

アプリケーションソフトウェアを実行するアプリケーション実行部を備え、前記言語情報保持部に保持されている使用言語情報が変更された後、前記アプリケーション実行部がアプリケーションソフトウェアを起動するとき、前記ダウンロード処理部は、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認する、

ことを特徴とする請求項1から3のいずれかに記載の情報処理装置。

#### 【請求項5】

前記ダウンロード処理部がデータ領域を確保できないことを確認した場合であっても、前記アプリケーション実行部は、アプリケーションソフトウェアを起動して、ユーザがアプリケーションを利用可能な状態とする、

ことを特徴とする請求項4に記載の情報処理装置。

#### 【請求項6】

20

情報処理装置におけるダウンロード処理方法であって、当該情報処理装置においてユーザにより選択されている使用言語を特定する使用言語情報を保持するステップと、アプリケーション本体と複数言語の言語依存ファイルを含むアプリケーションソフトウェアのダウンロード開始前に、アプリケーション本体と、使用言語情報により特定される使用言語の言語依存ファイルのデータサイズをサーバに問い合わせるステップと、

使用言語情報により特定される使用言語ではない言語の言語依存ファイルのデータ領域については記憶装置に確保せず、アプリケーション本体と、使用言語情報により特定される使用言語の言語依存ファイルのデータ領域を記憶装置に確保するステップと、

アプリケーション本体および使用言語の言語依存ファイルをダウンロードして、確保した記憶装置のデータ領域に記憶するステップと、

アプリケーションで使用する言語に変更が生じた場合に、変更後の使用言語の言語依存ファイルのデータサイズをサーバに問い合わせ、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認するステップと、

を有することを特徴とするダウンロード処理方法。

#### 【請求項7】

コンピュータに、

当該コンピュータにおいてユーザにより選択されている使用言語を特定する使用言語情報を保持する機能と、

アプリケーション本体と複数言語の言語依存ファイルを含むアプリケーションソフトウェアのダウンロード開始前に、アプリケーション本体と、使用言語情報により特定される使用言語の言語依存ファイルのデータサイズをサーバに問い合わせる機能と、

使用言語情報により特定される使用言語ではない言語の言語依存ファイルのデータ領域については記憶装置に確保せず、アプリケーション本体と、使用言語情報により特定される使用言語の言語依存ファイルのデータ領域を記憶装置に確保する機能と、

アプリケーション本体および使用言語の言語依存ファイルをダウンロードして、確保した記憶装置のデータ領域に記憶する機能と、

アプリケーションで使用する言語に変更が生じた場合に、変更後の使用言語の言語依存ファイルのデータサイズをサーバに問い合わせ、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認する機能と、

40

50

を実現させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ゲーム装置などの情報処理装置で実行される技術に関する。

【背景技術】

【0002】

従来よりゲームソフトウェアは、光ディスクや光磁気ディスク、ブルーレイディスクなどのROM媒体の形態で流通、販売されてきた。データ通信の高速化にともない、最近ではサーバがインターネット経由でゲームソフトウェアのイメージファイルを配信することも一般的となっている。 10

【0003】

ゲームソフトウェアは、起動ファイル、ゲームプログラムなどのゲームを実行するためのリソースファイル群、およびゲーム装置のオペレーティングシステム(OS:Operating System)が使用するファイル群を含んでいる。近年、ゲームソフトウェアに含まれるファイル数は多くなり、データサイズは大規模化する傾向にある。

【0004】

特許文献1はゲームソフトウェアを複数のグループに分割し、そのうち1つのグループ(第1グループ)を、ゲームソフトウェアの起動に必要なプログラムファイルおよびデータファイルが属するように構成したグループ構造を開示する。特許文献1に開示された技術によると、第1グループに属する全てのファイルのダウンロードが完了すれば、後続の第2グループ以降のファイルをダウンロードしなくてもゲームソフトウェアを起動でき、第2グループ以降のグループに属するファイルについては、ユーザがゲームをプレイ中にバックグラウンドでダウンロードしている。これによりゲームを開始するまでにかかるユーザのダウンロード待ち時間を短くしている。 20

【先行技術文献】

【特許文献】

【0005】

【特許文献1】米国特許出願公開第2014/201209号明細書

【発明の概要】

【発明が解決しようとする課題】

【0006】

ゲームソフトウェアのデータサイズが大規模化した理由の一つに、音声データおよび画像データが複数言語に対応して作成されていることがある。以下、このようなデータを「言語依存」ファイル(または言語リソースファイル)と呼ぶが、言語依存ファイルのデータサイズは、ソフトウェア全体のデータサイズに対してかなりの割合を占めている。そのためゲームソフトウェアを情報処理装置の記憶装置にダウンロードする際に、全ての言語の言語依存ファイルのデータサイズ分のデータ領域を記憶装置に確保すると、かなりのデータ領域を消費してしまうことになる。 30

【0007】

そこで本発明は、言語依存ファイルを効率的にダウンロードする技術を提供することを目的とする。

【課題を解決するための手段】

【0008】

上記課題を解決するために、本発明のある態様の情報処理装置は、当該情報処理装置においてユーザにより選択されている使用言語を特定する使用言語情報を保持する言語情報保持部と、アプリケーションソフトウェアを記憶装置にダウンロードするダウンロード処理部と、を備える。ダウンロード処理部は、アプリケーションソフトウェアのダウンロード開始前に、アプリケーション本体と、言語情報保持部に保持されている使用言語情報により特定される使用言語の言語依存ファイルのデータ領域を記憶装置に確保してから、ダ 40

50

ウンロードを実行するものであって、アプリケーションで使用する言語に変更が生じた場合に、ダウンロード処理部は、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認する。

#### 【0009】

本発明の別の態様は、情報処理装置におけるダウンロード処理方法である。この方法は、当該情報処理装置においてユーザにより選択されている使用言語を特定する使用言語情報を保持するステップと、アプリケーションソフトウェアのダウンロード開始前に、アプリケーション本体と、使用言語情報により特定される使用言語の言語依存ファイルのデータ領域を記憶装置に確保するステップと、アプリケーション本体および言語依存ファイルをダウンロードして、確保した記憶装置のデータ領域に記憶するステップと、アプリケーションで使用する言語に変更が生じた場合に、変更後の使用言語の言語依存ファイルのデータ領域を記憶装置に確保できるか否かを確認するステップと、を有する。

10

#### 【0010】

なお、以上の構成要素の任意の組合せ、本発明の表現を方法、装置、システム、記録媒体、コンピュータプログラムなどの間で変換したものもまた、本発明の態様として有効である。

#### 【発明の効果】

#### 【0011】

本発明の情報処理技術によると、言語依存ファイルを効率的にダウンロードする技術を提供することが可能となる。

20

#### 【図面の簡単な説明】

#### 【0012】

【図1】本発明の実施例にかかる情報処理システムを示す図である。

【図2】情報処理装置の機能ブロックを示す図である。

【図3】ゲームソフトウェアのファイル構成の概念図を示す図である。

【図4】ゲームソフトウェアの具体的なファイル構成例を示す図である。

【図5】グループとファイルの関係の一例を示す図である。

【図6】グループファイルの一例を示す図である。

【図7】ダウンロード処理を実行するための機能ブロックを示す図である。

【図8】通知情報の例を示す図である。

30

#### 【発明を実施するための形態】

#### 【0013】

図1は、本発明の実施例にかかる情報処理システム1を示す。情報処理システム1は、情報処理装置10と、ネットワークサーバ5と、デジタルコンテンツを配信するコンテンツサーバ12とを備え、これらはインターネットやLAN(Local Area Network)などのネットワーク3を介して接続している。コンテンツサーバ12は、ゲームソフトウェアなどのデジタルコンテンツを保持し、情報処理装置10からの要求に応じて送信する。

#### 【0014】

アクセスポイント(以下、「AP」とよぶ)8は、無線アクセスポイントおよびルータの機能を有し、情報処理装置10は、無線または有線経由でAP8に接続して、ネットワーク3上のネットワークサーバ5、コンテンツサーバ12と通信可能に接続する。

40

#### 【0015】

情報処理装置10は、ユーザが操作する入力装置6と無線または有線で接続し、入力装置6はユーザの操作結果を示す操作情報を情報処理装置10に出力する。情報処理装置10は入力装置6から操作情報を受け付けるとOS(システムソフトウェア)やゲームソフトウェアの処理に反映し、出力装置4から処理結果を出力させる。情報処理システム1において情報処理装置10はゲームソフトウェアを実行するゲーム装置やパーソナルコンピュータなどの端末装置であり、入力装置6はゲームコントローラなど情報処理装置10に対してユーザの操作情報を供給する機器であってよい。ユーザは情報処理装置10のOSにログインすることで、OSやゲームソフトウェアを操作できる。

50

**【0016】**

ネットワークサーバ5は情報処理システム1の運営主体により保守、管理され、情報処理システム1のユーザに対してネットワークサービスを提供する。ネットワークサーバ5はユーザを識別するネットワークアカウントを管理しており、ユーザは、ネットワークアカウントを用いて、ネットワークサーバ5が提供するネットワークサービスにサインインする。ユーザは情報処理装置10からネットワークサービスにサインインすることで、コンテンツサーバ12からデジタルコンテンツの配信を受けることができる。なおコンテンツサーバ12が、ネットワークサーバ5のユーザ管理機能を備えてもよい。実施例において、デジタルコンテンツは様々な種類のアプリケーションソフトウェアであってよいが、以下では、特にデジタルコンテンツがゲームソフトウェアである場合について説明する。

10

**【0017】**

補助記憶装置2はHDD(ハードディスクドライブ)やフラッシュメモリなどの大容量記憶装置であり、USB(Universal Serial Bus)などによって情報処理装置10と接続する外部記憶装置であってよく、内蔵型記憶装置であってもよい。実施例において補助記憶装置2は、コンテンツサーバ12からダウンロードするゲームソフトウェアを記憶する記憶装置である。出力装置4は画像を出力するディスプレイおよび音声を出力するスピーカーを有するテレビであってよく、またコンピュータディスプレイであってもよい。出力装置4は、情報処理装置10に有線ケーブルで接続されてよく、無線接続されてもよい。

**【0018】**

入力装置6は複数のプッシュ式の操作ボタンや、アナログ量を入力できるアナログスティック、回動式ボタンなどの複数の入力部を有して構成される。撮像装置であるカメラ7は出力装置4の近傍に設けられ、出力装置4周辺の空間を撮像する。情報処理装置10はカメラ7の撮像画像からユーザを顔認証して、ログインさせる機能をもつ。

20

**【0019】**

図2は、情報処理装置10の機能ブロック図を示す。情報処理装置10は、メイン電源ボタン20、電源ON用LED21、スタンバイ用LED22、システムコントローラ24、クロック26、デバイスコントローラ30、メディアドライブ32、USBモジュール34、フラッシュメモリ36、無線通信モジュール38、有線通信モジュール40、サブシステム50およびメインシステム60を有して構成される。

**【0020】**

30

メインシステム60は、メインCPU(Central Processing Unit)、主記憶装置であるメモリおよびメモリコントローラ、GPU(Graphics Processing Unit)などを備える。GPUはゲームプログラムの演算処理に主として利用される。これらの機能はシステムオンチップとして構成されて、1つのチップ上に形成されてよい。メインCPUは、補助記憶装置2やROM媒体44に記録されたゲームソフトウェアを実行する機能をもつ。

**【0021】**

サブシステム50は、サブCPU、主記憶装置であるメモリおよびメモリコントローラなどを備え、GPUを備えず、ゲームプログラムを実行する機能をもたない。サブCPUの回路ゲート数は、メインCPUの回路ゲート数よりも少なく、サブCPUの動作消費電力は、メインCPUの動作消費電力よりも少ない。サブCPUは、メインCPUがスタンバイ状態にある間においても動作し、消費電力を低く抑えるべく、その処理機能を制限されている。

40

**【0022】**

メイン電源ボタン20は、ユーザからの操作入力が行われるボタンであって、情報処理装置10の筐体の前面に設けられ、情報処理装置10のメインシステム60への電源供給をオンまたはオフするために操作される。電源ON用LED21は、メイン電源ボタン20がオンされたときに点灯し、スタンバイ用LED22は、メイン電源ボタン20がオフされたときに点灯する。

**【0023】**

システムコントローラ24は、ユーザによるメイン電源ボタン20の押下を検出する。

50

クロック26はリアルタイムクロックであって、現在の日時情報を生成し、システムコントローラ24やサブシステム50およびメインシステム60に供給する。

#### 【0024】

デバイスコントローラ30は、サウスブリッジのようにデバイス間の情報の受け渡しを実行するLSI (Large-Scale Integrated Circuit)として構成される。図示のように、デバイスコントローラ30には、システムコントローラ24、メディアドライブ32、USBモジュール34、フラッシュメモリ36、無線通信モジュール38、有線通信モジュール40、サブシステム50およびメインシステム60などのデバイスが接続される。デバイスコントローラ30は、それぞれのデバイスの電気特性の違いやデータ転送速度の差を吸収し、データ転送のタイミングを制御する。

10

#### 【0025】

メディアドライブ32は、ゲームなどのアプリケーションソフトウェアを記録したROM媒体44を装着して駆動し、ROM媒体44からプログラムやデータなどを読み出すドライブ装置である。ROM媒体44は、光ディスクや光磁気ディスク、ブルーレイディスクなどの読み出専用の記録メディアである。

#### 【0026】

USBモジュール34は、外部機器とUSBケーブルで接続するモジュールである。USBモジュール34は補助記憶装置2およびカメラ7とUSBケーブルで接続してもよい。フラッシュメモリ36は、内部ストレージを構成する補助記憶装置である。無線通信モジュール38は、Bluetooth(登録商標)プロトコルやIEEE802.11プロトコルなどの通信プロトコルで、たとえば入力装置6と無線通信する。有線通信モジュール40は、外部機器と有線通信し、AP8を介してネットワーク3に接続する。

20

#### 【0027】

図1に戻ってコンテンツサーバ12は、情報処理装置10にゲームソフトウェアを提供する。ゲームソフトウェアは、起動ファイル、ゲームプログラムなどのゲームを実行するためのリソースファイル群、および情報処理装置10のOSが使用するファイル群を含んでおり、コンテンツサーバ12は、本来ROM媒体44に記録されているゲームソフトウェアのイメージファイルを情報処理装置10に提供する。ゲームプログラムは、ゲームの実行に必要なプログラムであり、ゲームプログラムを走らせることでゲームが進行する。起動ファイルは、ゲームプログラムを起動するためのプログラムであり、起動ファイルを実行すると、ゲームプログラムが呼び出されて実行される。OSが使用するファイル群は、たとえば、情報処理装置10におけるメニュー画面に表示されるゲームアイコン画像などを含む。

30

#### 【0028】

ゲームソフトウェアはツリー型ディレクトリ構造を有し、最上層のルートディレクトリには起動ファイルが含まれている。下層のサブディレクトリは、ファイルの種類ごとに分類され、たとえば3Dモデル用のサブディレクトリ、テクスチャ用のサブディレクトリ、スクリプト用のサブディレクトリなどが形成されている。各サブディレクトリには、対応するファイルが含まれており、3Dモデル用サブディレクトリには複数の3Dモデルファイルが、テクスチャ用サブディレクトリには複数のテクスチャファイルが、スクリプト用サブディレクトリには複数のスクリプトファイルが含まれる。たとえばテクスチャ用のサブディレクトリには、ゲームのシーン1用のテクスチャファイル、シーン2用のテクスチャファイル、シーン3用のテクスチャファイルなどが含まれている。

40

#### 【0029】

最近のゲームは、複数言語に対応して作成されることが多い。そのため音声データおよび画像データが複数の言語ごとに作成され、複数言語の音声ファイルおよび画像ファイルが1つのパッケージソフトウェアに収められている。以下では、言語ごとに作成された音声ファイルおよび画像ファイルを「言語依存ファイル」と呼ぶ。

#### 【0030】

このような言語依存ファイルは、ゲームシナリオの長大化により、音声ファイルのデー

50

タサイズが非常に大きいことが知られている。そこで実施例のゲームソフトウェアは、ユーザが必要な言語依存ファイルのみをインストールできるように、言語ごとに音声ファイルおよび画像ファイルを集合させたりソースファイルをもつ形式を有する。この意味において、言語依存ファイルを、言語リソースファイルと呼ぶこともある。

【0031】

図3は、ゲームソフトウェアのファイル構成の概念図を示す。実施例のゲームソフトウェア70は複数のファイルによって構成され、図示されるように複数のグループ72に論理的に分割される。各ファイルは複数のグループ72のうち少なくとも1つのグループに属し、また各グループ72には少なくとも1つのファイルが属している。図3に示すゲームソフトウェア70には、先頭グループとして第1グループ72aが存在し、それに後続するグループとして第2グループ72b、第3グループ72c、第4グループ72d、第5グループ72e、第6グループ72fが存在している。なお第6グループ72fに後続する7番目以降のグループ72が存在していてもよい。各グループは、第1、第2などのグループ番号によって識別される。

【0032】

論理的に分割された各グループには、複数のサブディレクトリに含まれるファイルが属し、すなわち各グループは、種類の異なるファイルによって構成され、情報処理装置10がゲーム中のシーンやステージなどの特定の単位を実行するのに必要なファイルが属するように設定されている。

【0033】

第1グループ72aには、ゲームソフトウェア70の起動に必要なプログラムファイルおよびデータファイルが属している。したがって情報処理装置10は、ゲームソフトウェア70をコンテンツサーバ12から取得する場合、第1グループ72aに属する全てのファイルをダウンロードすれば、後続の第2グループ72b以降のファイルをダウンロードしなくとも、ただちにゲームソフトウェア70を起動することが可能となる。なお情報処理装置10は、第1グループ72aに属する全てのファイルを取得して、ゲームソフトウェア70を起動した後に、後続のグループ72に属するファイルをバックグラウンドでダウンロードする。このようにゲームの実行に必要な最低限のファイルをまず最初にダウンロードさせ、それらのファイルが揃った時点でゲームを実行可能とすることで、ユーザのダウンロード待ち時間を短くすることが可能となる。

【0034】

図4は、ゲームソフトウェアの具体的なファイル構成例を示す。第1グループ72aは、ゲームソフトウェア70の中で一番最初にダウンロードされるべきファイル群で構成されており、ここではゲームパラメータファイル、グループファイル、起動ファイルおよび必須リソースファイルが示されている。

【0035】

ここでゲームパラメータファイルは、情報処理装置10のOSが使用するファイルであり、たとえばタイトルIDやディスプレイ解像度などの情報、またアイコン画像データなどを含んでいる。

【0036】

グループファイルは、各ファイルがどのグループに含まれるかを記述する定義ファイルである。たとえばグループファイルはXMLで表現されてよいが、他のプログラム言語によって表現されてもよく、その形式は問わない。グループファイルについては、図5、図6に関して後述する。

【0037】

起動ファイルは、ゲームプログラムを起動するためのプログラムである。また必須リソースファイルは、ゲーム実行に必須となるプログラムなどのリソースファイルやゲーム全体で使用する共通ファイルなどを含む。

【0038】

情報処理装置10は、ゲームソフトウェア70をコンテンツサーバ12などからダウン

10

20

30

40

50

ロードする場合、第1グループ72aに属するファイル群を全て取得してインストールすれば、ゲームを起動できる。逆に言えば、第1グループ72aは、ユーザがゲームの一部をプレイするために必要なファイル群を含むように構成されている。なお、ここでいうゲームプレイは、たとえばユーザがキャラクタを決定したり、ゲームレベルを決定するなど、ゲーム開始時に行う設定行動も含むものであってよい。つまり第1グループ72aは、ゲームを起動し、ユーザが少なくとも何らかの動作を行える状態にするために必要なファイル群を含んで構成されている。第1グループ72aに含まれるファイル群を用いて実行可能となるゲームプレイは、たとえばゲームの初期設定だけであってもよく、またゲームの第1ステージまでプレイ可能とするものであってもよい。これはゲームメーカ次第である。

10

#### 【0039】

図4に示す例では、第2グループ72bには、シーン1用の複数のリソースファイルが属しており、第3グループ72cには、シーン2用の複数のリソースファイルが属しており、第4グループ72dには、シーン3用の複数のリソースファイルが属している。具体的に複数のリソースファイルは、プログラム、特定のシーン用の3Dモデルファイル、テクスチャファイル、スクリプトファイルなどを含み、ディレクトリ構造の複数のサブディレクトリに含まれるファイルを含む。

#### 【0040】

図5は、グループとファイルの関係の一例を示す。ここでは、ファイルA～Nが、各グループ72に属していることが示される。図示されるように、各ファイルは複数のグループ72のうち少なくとも1つのグループに属し、また各グループ72には少なくとも1つのファイルが属している。なおファイルGは、第2グループ72b、第3グループ72cおよび第4グループ72dに属している。このことは、ファイルGが、ゲーム中のシーン1、シーン2、シーン3を構成する上で必要なファイルであることを意味し、このように1つのファイルが、複数のグループに所属することもある。なお同様にファイルKも、複数のグループ72、すなわち第4グループ72dおよび第5グループ72eに属している。

20

#### 【0041】

図6は、グループファイルの一例を示す。既述したように、グループファイルはXMLによって表現されてもよく、他のプログラム言語によって表現されてもよい。図6には、理解を容易にするために、グループとファイルの対応関係をテーブル形式で表現したグループファイルを示している。情報処理装置10は、ゲームソフトウェア70の各ファイルをダウンロードする際に、グループファイルを参照して、あるグループに属するファイルが全て揃ったか、または揃っていないかを判定することが可能となる。たとえば第1グループ72aに関して言えば、情報処理装置10はグループファイルを参照することで、第1グループ72aに属するファイルが、ファイルA, B, C, D, E, Fであることを認識できるため、これらのファイルが補助記憶装置2に記憶されていれば、第1グループ72aに属するファイルが全て揃ったことを判定する。なおグループファイルは、ファイルA～Fのうちのいずれか1つであってよい。

30

#### 【0042】

このようにゲームソフトウェア70が複数のグループにより構成されることで、情報処理装置10は、グループのダウンロードの優先順位を定めるダウンロード順序にしたがって、ファイルをダウンロードすることができる。ゲームソフトウェア70が、1人でプレイするシングルプレイ用のリソースファイルと、複数人でプレイするマルチプレイ用のリソースファイルを含む場合、ユーザがシングルプレイを望む場合には、まず第1グループ72aをダウンロードした後、シングルプレイ用のグループを優先的にダウンロードし、シングルプレイを楽しみながら、バックグラウンドでマルチプレイ用のグループをダウンロードする。一方で、ユーザがマルチプレイを望む場合には、まず第1グループ72aをダウンロードした後、マルチプレイ用のグループを優先的にダウンロードし、マルチプレイを楽しみながら、バックグラウンドで、シングルプレイ用のグループをダウンロードする。

40

50

## 【0043】

ダウンロード順序において、最初にダウンロードするべきグループが第1グループ72aであることは、デフォルトで設定されている。その後のダウンロードグループについてはゲームによって指定されてよく、上記したようにユーザによりシングルプレイが選択された場合には、ゲームが、ゲーム進行順序にしたがったシングルプレイ用のグループのダウンロード順序を指定する。これにより効率的なゲームソフトウェア70のダウンロードが実現される。

## 【0044】

情報処理装置10では、ユーザにより使用される言語の情報が設定されている。たとえばユーザが日本人であれば、使用される言語が日本語に設定され、ユーザが米国人であれば、使用される言語は英語に設定される。情報処理装置10のOSは、ユーザにより選択された使用言語情報を保持し、ホーム画面などOSが提供する画面は、使用言語情報にもとづいて生成される。

10

## 【0045】

実施例の情報処理装置10は、コンテンツサーバ12からのゲームソフトウェアのダウンロード処理に際して、設定されている使用言語情報に対応する言語依存ファイル（言語リソースファイル）のみをダウンロードする。そのため使用言語が英語に設定されれば、情報処理装置10は、英語リソースファイルをダウンロードするが、他の言語のリソースファイルはダウンロードしない。

## 【0046】

20

図7は、情報処理装置10におけるダウンロード処理を実行するための機能ブロックを示す。メインシステム60は、言語情報保持部100、受付部102、ダウンロード処理部110およびアプリケーション実行部120を備える。ダウンロード処理部110は、データ領域確保部112、ダウンロード実行部114および通知部116を有して、ゲームソフトウェアを補助記憶装置2にダウンロードする処理を実施する。アプリケーション実行部120は、ダウンロードされたゲームソフトウェアを実行する。

## 【0047】

これらの構成は、ハードウェアコンポーネントでいえば、任意のコンピュータのCPU、メモリ、メモリにロードされたプログラム、ストレージなどによって実現されるが、ここではそれらの連携によって実現される機能ブロックを描いている。したがって、これらの機能ブロックがハードウェアのみ、ソフトウェアのみ、またはそれらの組合せによっていろいろな形で実現できることは、当業者には理解されるところである。

30

## 【0048】

言語情報保持部100は、情報処理装置10においてユーザにより選択されている使用言語を特定する使用言語情報を保持する。使用言語は、情報処理装置10の初期設定時にユーザにより選択されるが、ユーザは初期設定した使用言語を自由に変更できる。ユーザにより選択された使用言語は、「システム言語」とも呼ばれ、システム言語は、情報処理装置10のOSが提供するシステム画面を構成する言語となる。たとえばシステム言語が英語である場合、OSはシステム画面を英語で生成することになる。

## 【0049】

40

ゲームソフトウェア70のダウンロード処理において、補助記憶装置2は、ゲームソフトウェア70を構成する複数のファイルを格納するための記憶装置として利用される。ゲームソフトウェア70において、各ファイルは少なくとも1つのグループに所属しており、また各グループには少なくとも1つのファイルが所属しているが、ダウンロード処理はグループ単位で実行される。たとえば、ファイルX, Y, ZがグループSに属している場合、グループSのダウンロード要求が生成された場合には、ファイルX, Y, Zがコンテンツサーバ12からダウンロードされて、グループSに属する全てのファイルX, Y, Zが補助記憶装置2に記憶されるようになる。なお既にファイルXがダウンロード済みである場合には、ファイルY, Zがコンテンツサーバ12からダウンロードされて、これによりグループSに属する全てのファイルX, Y, Zが補助記憶装置2に記憶されるようにな

50

る。

#### 【0050】

情報処理装置10における具体的なダウンロード手順について説明する。

ユーザによるゲームソフトウェアの購入に際し、受付部102がゲームソフトウェアの販売ページのアドレスを受け取り、出力装置4に販売ページを表示させる。ユーザが入力装置6を操作して所望のゲームソフトウェアを選択すると、ダウンロード処理部110が、選択されたゲームソフトウェアを補助記憶装置2にダウンロードする処理を実施する。

#### 【0051】

具体的にダウンロード処理部110において、データ領域確保部112が、ゲームソフトウェアのダウンロード開始前に、ゲーム本体と、言語情報保持部100に保持されている使用言語情報により特定される使用言語の言語依存ファイルのデータサイズを、コンテンツサーバ12に問い合わせる。ここでゲーム本体は、ダウンロード対象となるゲームソフトウェアから、全ての言語リソースファイルを除いたファイル群で構成される。したがってゲーム本体のデータサイズは、ゲームソフトウェアの全体のデータサイズから、全言語リソースファイルのデータサイズを減算した値となる。実施例では、使用言語が英語に設定されているものとし、したがってデータ領域確保部112は、ゲーム本体のデータサイズと、英語の言語依存ファイル（英語リソースファイル）のデータサイズを、コンテンツサーバ12に問い合わせる。コンテンツサーバ12は、それぞれのデータサイズ、またはデータサイズの合計値を情報処理装置10に回答する。

#### 【0052】

コンテンツサーバ12からの回答を受けて、データ領域確保部112は、ゲーム本体と、英語リソースファイルのデータ領域を、補助記憶装置2に確保できるか否かを確認する。このときデータ領域確保部112は、連続領域の所定のサイズを最小単位として、ゲーム本体と、英語リソースファイルのデータ領域を確保できるか否かを確認し、確保できる場合には、そのデータ領域を、当該ゲームソフトウェア用に確保する。データ領域を確保するとは、他の用途に使用しないことを意味する。たとえば連続領域の最小単位は128Mバイトであってよい。

#### 【0053】

実施例において、データ領域確保部112は、言語リソースファイルに関し、言語情報保持部100に保持された使用言語のリソースファイルのデータ領域のみを確保し、他の言語のリソースファイルのデータ領域の確保を行わない。一般にユーザは、使用言語を変更することが少なく、他の言語のリソースファイルのデータ領域を補助記憶装置2に確保しておくと、結果として、その領域が無駄になることがある。そのため実施例では、データ領域確保部112が、言語リソースファイルに関し、使用言語のリソースファイルのデータ領域のみを確保することで、補助記憶装置2のデータ領域を有効利用する。データ領域確保部112が、ゲーム本体および英語リソースファイルのデータ領域を補助記憶装置2に確保してから、ダウンロード実行部114は、ゲーム本体および英語リソースファイルをダウンロードして、確保した補助記憶装置2のデータ領域に記憶する。

#### 【0054】

実施例のダウンロード処理によれば、ダウンロード実行部114が、第1グループ72aに属する全てのファイルをダウンロードすれば、後続の第2グループ72b以降のファイルをダウンロードしなくても、アプリケーション実行部120が、ただちにゲームソフトウェアを起動することが可能となる。なおダウンロード実行部114は、引き続き後続のグループに属するファイルをバックグラウンドでダウンロードする。

#### 【0055】

このようにユーザがゲームソフトウェアを購入すると、ダウンロード処理部110が、ゲーム本体および使用言語リソースファイルをダウンロードし、ユーザがゲームをプレイできるようになる。

#### 【0056】

その後、アプリケーションで使用する言語に変更が生じることがある。使用言語に変更

10

20

30

40

50

が生じる第1のケースは、言語情報保持部100に保持されている使用言語情報が変更されたとき、つまりユーザがシステム言語を変更したときに発生する。また使用言語に変更が生じる第2のケースは、ゲーム内でユーザが使用言語を変更したときに発生する。

#### 【0057】

アプリケーションで使用する言語に変更が生じた場合、データ領域確保部112は、変更後の使用言語のリソースファイル（言語依存ファイル）のデータサイズをコンテンツサーバ12に問い合わせ、変更後の使用言語のリソースファイルのデータ領域を補助記憶装置2に確保できるか否かを確認する。このときデータ領域確保部112は、128Mバイトの連続領域を最小単位として、補助記憶装置2にデータ領域を確保できるか否かを確認する。ここでデータ領域確保部112が、データ領域を確保できないことを確認すると、通知部116が、ダウンロードエラーに関する通知を出力装置4に表示する。

10

#### 【0058】

上記した第1のケース、つまりユーザがシステム言語を変更した後に、アプリケーション実行部120がゲームソフトウェアを起動するとき、データ領域確保部112は、言語情報保持部100に保持されている使用言語情報が変更されたことを確認して、変更後の使用言語のリソースファイルのデータ領域を補助記憶装置2に確保できるか否かを確認する。たとえばユーザがシステム言語を英語から日本語に変更した場合、データ領域確保部112は、日本語リソースファイルのデータ領域を、補助記憶装置2に確保できるか否かを確認する。

#### 【0059】

20

なおアプリケーション実行部120は、この確認結果によらず、ゲームソフトウェアを起動して、ユーザがゲームを利用可能な状態、つまりプレイ可能な状態としてよい。データ領域確保部112が日本語リソースファイルのデータ領域を確保できないことを確認した場合であっても、変更前の英語リソースファイルはインストールされているため、アプリケーション実行部120は、英語リソースファイルを用いて、ゲーム本体を実行してよい。

#### 【0060】

図8は、通知部116により表示される通知情報130を示す。データ領域確保部112が日本語リソースファイルのデータ領域を補助記憶装置2に確保できない場合、通知部116が、ダウンロードエラーに関する通知情報130を表示する。ユーザはゲーム画面上で通知情報130を視認することで、ゲーム音声等が英語のままで、日本語に切り替わらないことを認識する。

30

#### 【0061】

以上の実施例は例示であり、それらの各構成要素や各処理プロセスの組合せにいろいろな変形例が可能のこと、またそうした変形例も本発明の範囲にあることは当業者に理解されるところである。実施例では、アプリケーションの例としてゲームを示したが、それ以外のアプリケーションであってもよい。

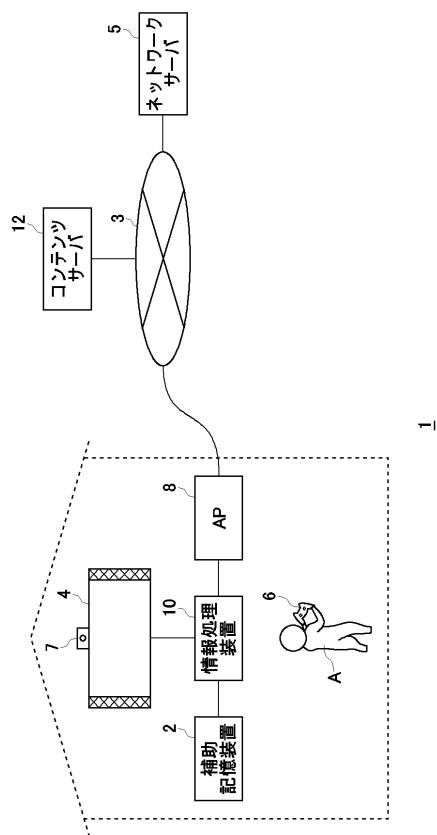
#### 【符号の説明】

#### 【0062】

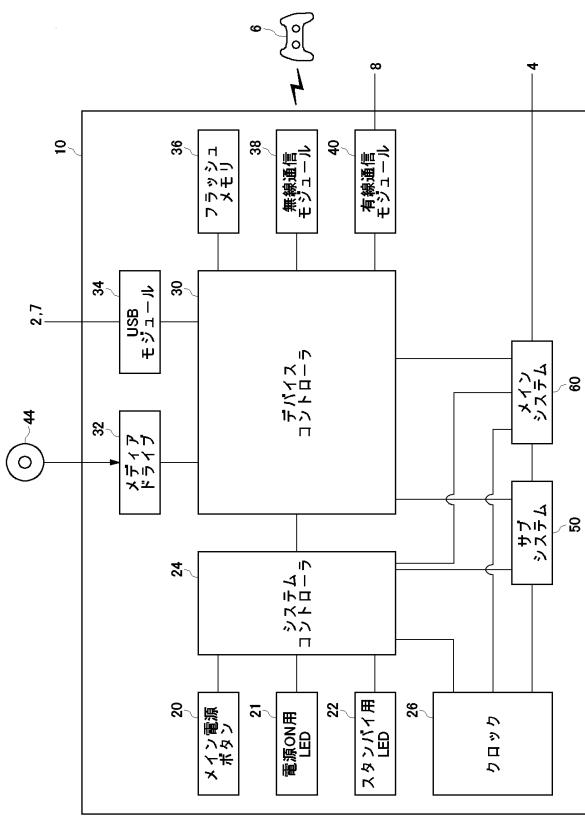
1・・・情報処理システム、2・・・補助記憶装置、10・・・情報処理装置、12・・・コンテンツサーバ、100・・・言語情報保持部、102・・・受付部、110・・・ダウンロード処理部、112・・・データ領域確保部、114・・・ダウンロード実行部、116・・・通知部、120・・・アプリケーション実行部。

40

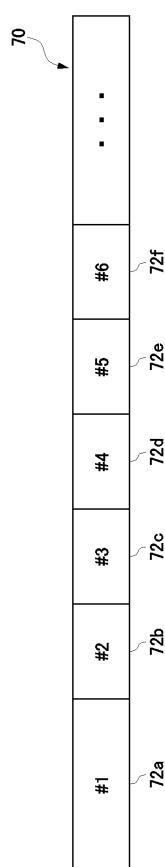
【 図 1 】



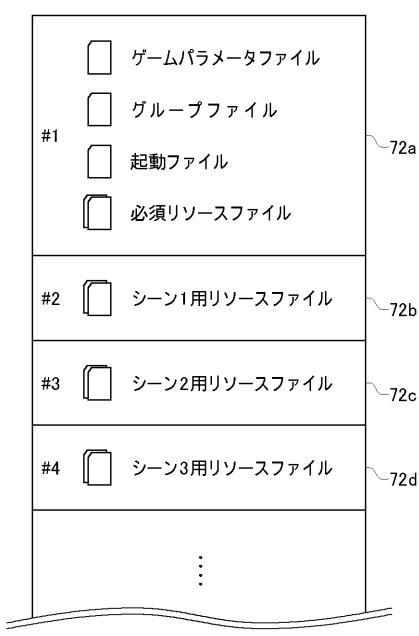
【 図 2 】



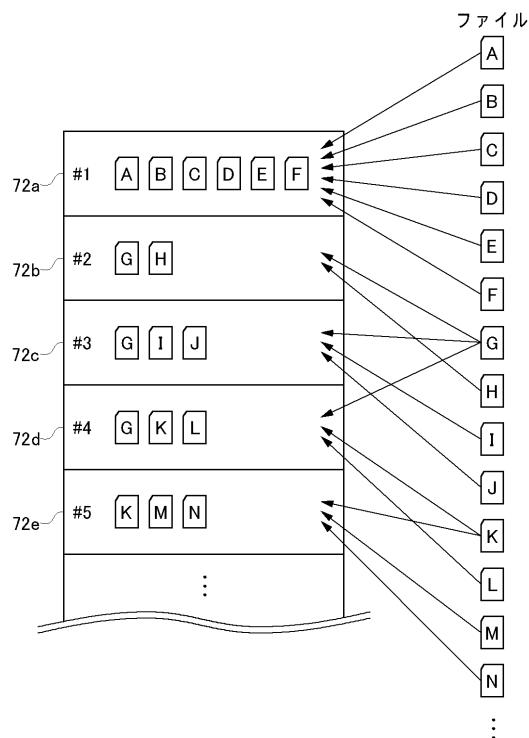
【図3】



【図4】



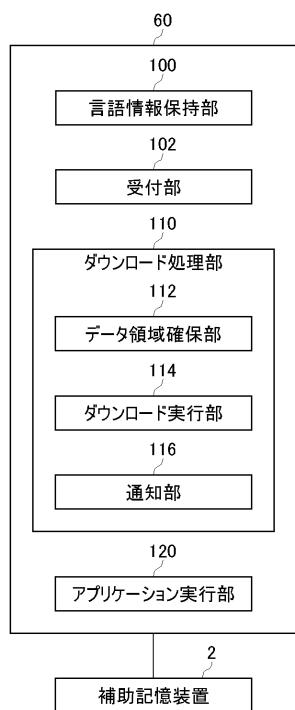
【図5】



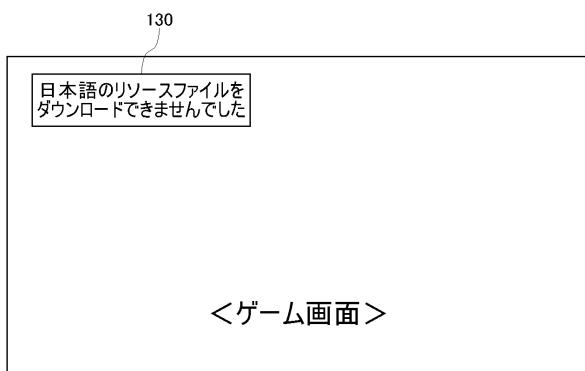
【図6】

72a	グループ	#1	#2	#3	#4	#5	...
72b	ファイル	A,B,C,D,E,F	G,H	G,I,J	G,K,L	K,M,N	...

【図7】



【図8】



---

フロントページの続き

(72)発明者 高橋 正貴

東京都港区港南1丁目7番1号 株式会社ソニー・インタラクティブエンタテインメント内

(72)発明者 馬場 俊介

東京都港区港南1丁目7番1号 株式会社ソニー・インタラクティブエンタテインメント内

審査官 岩田 玲彦

(56)参考文献 国際公開第2014/111984 (WO, A1)

国際公開第2005/083575 (WO, A1)

国際公開第2008/149501 (WO, A1)

(58)調査した分野(Int.Cl., DB名)

G 06 F 13 / 00

A 63 F 13 / 358

G 06 F 16 / 182