

## 第 9 部

# 利用者情報管理



## 第 1 章

### はじめに

現在、コンピュータのネットワーク分野は大きな発展を遂げ、その規模は、1つの地域内で接続されたローカルエリアネットワークから、専用回線で接続された広域ネットワークへと、今や広範に渡るようになってきている。こうした条件の下で、情報の交換や通信が頻繁に行われているわけである。ここで、情報に関して取り上げてみると、情報はネットワークの拡大に伴ってますます増大し、有効的に、そして、容易に情報を活用することが難しくなってくる。したがって、こうした現状を鑑みると、もともと一地域内で管理していた情報というものも、地理的あるいは管理的に異なった地域を踏まえて、円滑的に利用できるものでなければならない。情報はその性質上ローカルに管理されるものであるから、外部に負担をかけずに利用できるべきである。

一方、今までのローカルエリアネットワークも、それに接続されるコンピュータの増加に伴い独立に発達しているので、管理体系も複雑、個性化している。つまり、こうしたネットワークは独自のアドレス体系、名前サービスなどを持っている。広域分散環境においては、こうした独立に発達したネットワーク同志を接続する上で、このような機能をサポートすることが重要になってくる。言い換えれば、情報を提供したり受け取ったり、交換したりする必要性が当然増えてくるわけであるから、特に情報を管理する機構が必要になってくる。また、その管理もより有効的な、かつ能率的な、ネットワークへの負荷の少ないものが望ましい。

このようなことを踏まえて、本研究では、情報の中で特に、ユーザ個人の情報に的を絞って、このユーザ情報の確保、提供、必要に応じた収集、交換といった一連のユーザ情報の管理システムについて研究を行った。

## 第 2 章

### 本研究の目的と背景

#### 2.1 本研究の目的

ネットワークの相互接続が進むにつれ、それらのネットワーク同志で新たに必要な情報が出てくる。言い方を変えれば、あるネットワークが別のネットワークと接続していれば、そのネットワークに関わるユーザの情報も当然必要な場合が出てくる。例えば、別のネットワークに属するユーザに必要な情報を送りたい場合、あるいは、直接そのユーザにアクセスしたい場合、さらには、情報を提供してもらいたい場合である。このとき、ネットワーク単位ではなく、ユーザ単位の情報を得られることが不可欠である。それは単に位置情報に留まらず動的な情報も含めて、である。こうしたユーザ情報はユーザに負担をかけずに提供されることが望まれる。

広域分散環境のもとでのユーザ数を考えると、そのユーザ情報量がかなり多くなることは確かで、こうしたユーザ情報を一括して管理するシステムがあれば、効率的に自由な情報交換をすることが可能になる。ユーザ情報を管理できるようなシステムを作ることによってネットワークの状態の動的変化にも対応できるわけである。

これらをまとめるとユーザ情報管理システムは

- ユーザに負担をかけない情報の提供
- ユーザに関する情報をサポートする機能
- 広域分散環境への拡張性、発展性を十分考慮したもの

といった背景のもとに構築された。

## 2.2 定義

本研究を論ずるに当たって、必要な語句を以下のように定義づけることにした。

- ドメインとは1つ又は複数のドメイン又はシステムを包含したものである。
- システムとはネットワーク上の最小のアドレスづけの単位であり、OS又はこれに準ずるソフトウェアで一括して管理されるユーザファイル、ディレクトリ、サービスなどの資源全体である。ユーザ情報管理システムとは、これとは別にユーザ情報管理に関する資源全体を総称したものである。
- 広域分散環境のユーザに対してネットワークレベルで提供されるアプリケーションをサービスといい、サービスはシステム又はドメインにサービス名とそのサービスを行う実体(サービスプログラム)の位置(システムのインターネットアドレスとポート番号)である。

## 2.3 ユーザ情報の分類

ユーザ情報を管理するに当たって、ユーザの情報の性質を理解する必要がある。下位レベルの話は保留しておくが、上位レベルでいえば、アクセスしているホスト名から、ttyの種類、作業の状態まで、情報には様々な種類がある。こうしたユーザ情報はファイル保存の立場に基づくと大きく2つに分けることができる。

- static な情報
- dynamic な情報

static な情報とは、ユーザ名、ユーザアドレスといった固定的であり、ユーザからの特別な更新がある場合を除けば変更されることのない情報である。この更新も、ハードウェアレベルで(例えば、新たに他のネットワークと接続した場合や、ローカルネットワーク内での必要に応じた更新)行なわれる場合と、ユーザ個人の要求に基づいて能動的に行なわれる場合の2つである。したがって、こうした特別な場合を除けば書き換えが行なう必要がないので、この情報は、ユーザごとにまとめてデータベースに保存することができ、照会要求が来た時にその要求に応じてデータベースを参照すればよい情報である。

一方、dynamic な情報とは、ホスト名、tty、idle time といった刻々と変化し得る情報である。これは static な情報と違って、データベースに保存しようとするれば、頻繁に情報の変更、削除、投入が行なわれる。言い換えれば、この情報が常に信頼性のある情報であるためには、常に変化が生じたら、データベースを更新することが必要である。さらにその更新は当然ユーザに負担をかけないものでなければならない。情報を格納するにはデータベースに依らなければならないので、その情報を取り込む時期、方法、更新の方法等を、十分考慮しなければならない。

## 第 3 章

### ユーザ情報管理体系

#### 3.1 ユーザ情報管理システムにおける 4 つの機能

ユーザ情報の性質を理解した上で、その管理システムを円滑的に、そして能率的に構築していく必要がある。忘れてはならないことはこのシステムはローカルネットワークのみならず、今後も成長し続ける広域分散環境に耐え得るものでなければならない。これを踏まえて、ユーザ情報管理システムの仕組みをいくつかの基本的な動作を分析することにより概観する。情報を扱う点でユーザ情報管理システムには大きく分けて 4 つの機能がある。

- 問い合わせに対する応答機能
- 利用者個人情報の検索機能
- 利用者個人情報の収集と格納機能
- 利用者個人情報データベースの更新と整合機能

以下、これら 4 つの機能について述べる。

##### 3.1.1 問い合わせに対する応答機能

情報の問い合わせは情報を必要とする時に起こる。その問い合わせはユーザによって コマンド等の形で要求されたり、あるいは、アプリケーションを実行した際にその プロセスの過程で要求される。いずれにせよ、問い合わせはクライアントプロセス から発せられる。こうしたクライアントプロセスからの照会要求により、必要な ユーザ情報を提供する。その情報は要求しているところへ確実に転送されなければならない。また、要求に迅速に対処しなければならない。基本的には、特定のサーバがその役割を果たす。サーバは照会要求がどこで生じ、その要求の内容は何であるか、さらにはどこに提供するかを保証する。

### 3.1.2 利用者個人情報の検索機能

クライアントから発せられる照会要求は、特定のユーザに関する情報である場合と、多数のユーザの特定の情報である場合とがある。ユーザを特定している場合は、それがどのユーザの情報であるか、また、情報を特定している場合には、それが何の情報であるのかを検索する。照会要求の対象は1つあるいは複数の限定された情報であり、同時に1人あるいはそれ以上のユーザを対象とするものである。全てのユーザ情報の中からその要求に該当するものを検索し、選出することが必要である。

### 3.1.3 利用者個人情報の収集と格納機能

クライアントへ情報を提供するに先立ち、管理下のユーザの情報を保存しておくことは当然である。そしてその情報は、ユーザ情報に関する全ての要求をサポートできるものでなければならない。そのためにはユーザの現時点での新しい情報を常に認識しておくべきである。したがって、情報を適宜収集し、データベースに格納する。格納については、ユーザごとにそして種類別に整理しておくことが、検索する上でも必要である。ここで、情報を収集するにあたっては、信頼性を十分考慮し、収集の時期とその過程について検討しなければならない。

### 3.1.4 利用者個人情報データベースの更新と整合機能

ユーザ情報が常に新しいものであるためには、データベースの更新が行なわれなければならない。その更新も、どのユーザに関してか、そして、どの情報に関してかを判別し、正確に行なわれる必要がある。また、不要な更新はさげなければならない。あるいは、古い情報の中でも不必要な情報は、データベースが無駄に大きくならないよう捨てなければならない。ネットワーク内、あるいはドメインに関わる利用者とその情報の登録、変更、抹消は必要に応じて適宜行なわれなければならない。さもなければ、既に存在しない情報がクライアントから要求される場合が生じるし、また、ユーザ情報データベースに不整合が生じる。この場合、整合要求によりデータベースの整合が行われる。

## 3.2 ユーザ情報管理システムを支える3つの基本サーバ

ユーザ情報管理システムには基本的に4つの機能があることを先に述べたが、これらの機能を有効に働かせるためには3つの基本サーバが必要である。1つはクライアントの要求を受け、必要な情報を検索し提供する



サーバ、そして、もう1つは情報を収集しデータを更新するサーバ、さらには、広域分散環境を考慮し、先のサーバの所在を示すサーバである。これらのサーバを以下のように名付けることにする。

- user server
- information server
- domain server

これら各々のサーバが相互に働きあって全体の管理システムを形成している。まず、user server は各ユーザの情報をそのデータベースに格納し管理している。information server は、時間的に変化しやすい dynamic な情報を特に管理するサーバでありこれを user server に報告する。domain server は各ユーザに関するドメイン名と user server との対応づけを行っている。次章でそれぞれのサーバの役割と機能を示す。

## 第 4 章

### user server

#### 4.1 user server の役割とその機能

user server は各ユーザの情報を自らのデータベースの中に格納しておき、クライアントからの照会要求に応じてその情報を提供する。データベースにはユーザの static な情報と、dynamic な情報とが保存されており、各ユーザ毎に、そして情報の種類ごとに整理されている。つまり、ユーザ情報はすべてこの user server のデータベースに存在する。言い換えれば、user server はその管理下におかれている全てのユーザの情報をデータベースに持っている。したがって、要求の対象となるユーザの情報を選出する役割も同時に果たしている。また、更新要求が発せられると、データベースの書換えを実行する (この更新は static な情報に関してである)。又、ユーザの新規登録、抹消などに関しても同様である。

一方、クライアントプロセスからの要求に答えるという側面の他に、dynamic な情報などを取り込むという側面もある。送られた情報を取り込んで、各ユーザ単位のデータベースに仕分けし、格納する役割もある。つまり、この user server はユーザ情報管理システムにおいて中心的な役割を果たすと同時に重要な機能である。

#### 4.2 user server の通信モデル

user server の役割は明確化されたが、user server はどのくらいの範囲でユーザ情報を管理するか、つまりどこまでのユーザをその管理下に置くかが問われる。

user server を構築するにあたっては、user server の起動、負荷を考慮した様々なモデルを想定した。なぜなら、サーバは必要最小限に、かつ効率的にその役割を果たせれば良いからである。

まず、user server を各システムに 1 つおいた場合 (図 1 を参照) であるが、これは、クライアントプロセスから次々に送られてくる要求全てに

同時に答えることが困難な上に、情報を取り込む場合にも同じようなことがいえる。唯一つのサーバではかなりの負荷がかかってしまうのである。

次に、user server を各ユーザと 1 対 1 に対応させた場合 (図 2 を参照) には、クライアントの要求に同時に答えることは可能であるが、可能な限り起こり得る全てのクライアントからの照会要求が生じない限り、それだけ多くの user server を起動させることは無駄であり、通信の混乱を引き起こしやすい。

サーバの負荷、最小限の起動、といったことを満足させるために次のような通信モデルを構築した。(図 3 を参照)

まず、parent user server(PUS) を各システムに 1 つ置き、この PUS はクライアントからの要求に応じて (例えば A というユーザの情報を要求しているクライアントがあれば) fork して専用の child user server(CUS) とし (例えば A user server)、その CUS に仕事を委任する。これにより、必要な際に PUS は CUS を起動させれば良いし、複数のクライアントからの要求に同時に答えることが可能になる。したがって全体の user server にかかる負荷も最小限に抑えられる。

ここで user server は 2 つの構成要素を持つ

- parent user server(PUS)
- child user server(CUS)

PUS はシステムに 1 つ存在し、ユーザ情報要求がくると、それがどのユーザに関してかを認識し、そのユーザに関しての情報を管理する CUS を起動させる。PUS はクライアントからの要求を受けるだけで、直接それに答えることはしない。情報を取り込む場合にも、どのユーザのデータベースに書き込まれるかを判断し、あとはそのデータベースを管理する CUS に任せる。つまり、PUS はクライアントからの受け口であり、その要求に見合った CUS を起動させるのである。

CUS は各ユーザに対して 1 つ存在すると思ってよい。しかし CUS の起動は PUS に委ねられている。CUS は自分の管理すべき 1 人のユーザ情報をデータベースに持ち、クライアントへ必要な情報を直接提供する。また、更新要求があれば、自分のデータベースを更新する。つまり、CUS はユーザ情報の格納、更新そして、クライアントとの通信を行なっている。

### 4.3 データベースのフォーマット

データベースに保存される情報はデータとして、

\* static 情報…本名、ローカル名、インターネットアドレス phone number、

mail box

\* dynamic 情報 … ホスト名、tty、w 情報 (login time, idle time, what) をサポートすることにする。ここで what とはユーザのホスト上での状態のことである。このデータは、検索、更新を実行し易いように各ユーザごとに存在する。従ってこれらのデータのヘッダには、ユーザ名と、それに加えて各情報ごとのラベル付けが記されている。

## 第 5 章

### information server

#### 5.1 information server の役割とその機能

user server によって管理されているユーザ情報のうち特に dynamic 情報が信頼性の高いものであるためには、常に変化が生じるごとに書き換えが必要である。そのような情報を user server がいちいち収集するのでは能率が悪い。つまり、情報を収集するという負担を user server にかかせないために、それを実行してくれる特別なサーバが必要になってくる。information server はこうした機能をサポートしてくれる。データベースへの格納は user server に任せることにして、information server は情報を集めて、user server に報告することを役割とする。

#### 5.2 information server の通信モデル

この information server が情報を収集する方法としてはいくつかのモデルが考えられる。まず、information server を各システムに一つ置き、各ホストに情報を転送してもらう場合 (図 5 を参照)、この information server は各ホストにアクセスしているユーザの情報を取り込むわけであるが、ホストの情報を一度に取り込むような場合が生じ、server にかかる負荷も大きくなるし、データベースに一度保存しておくにしてもそのファイルは膨大なものになってしまう。さらに重大なこととして、ホストが情報を提供できるためにはクライアントプロセスによらなければならないので、ユーザに負担をかけることに他ならない。

また、各ホストに information server が情報を聞きに行くにしても (図 6 を参照)、ホストはそもそも簡単に増やすことができるので、ホストが増えるたびに information server が聞きに行くホストを information server のデータベースにいちいち登録しなければならない。小規模なネットワークであれば対応できるが、広域分散環境下においては不便である。

最終的に、information server は各ホストと 1 対 1 に対応するよう、ホスト上で常に起動させておく必要がある。(図 7 を参照)

なぜなら、dynamic 情報はいつなごき変化するか予測が付き難いからであるし、また、ホストが増加しても、そのホスト上で information server を起動し、user server の所在を示すだけで容易にホストがそのシステムに登録されるからである。各ホストの情報はそのまま user server に転送され、各々のホストにアクセスしているユーザごとに割り振られ、格納されるのである。

### 5.3 information server の報告する情報

各ホストに information server を置くことで、こういった種類の情報が user server に転送されるかを考える。問題なのは dynamic な情報が常にユーザサーバに取り込まなければ現時点でのユーザの正確な情報が得られないということである。

ここでまず、information server はデータベースを持つべきかどうか問われる。もし、定期的に information server が情報を取り込むのであれば、データベースを持つ必要がなく、information server が情報を収集すると同時に user server に転送すれば良い。しかし、その場合には、どのくらいの間隔でホストに情報を聞きに行くのか、その尺度が難しい。長い間隔では情報は信頼性に乏しいものになり、短い間隔では server 間の通信が多くなり、システムの負荷につながる。また、データに変化がなくても、通信が起こることは意味がない。

ここで、information server がデータベースを持つことにより、前後のデータを照合し、変化が生じた場合のみその変化分だけを user server に報告することができる。無駄な通信も避けられるわけである。しかし、データベースを持つと、ディスクへの書き込みが生じる。これが 1 システム内の全てのホストについてなされると、ファイルシステムの負荷につながる。情報の中で、最も変化するものをあげると、idle time、次に what 情報である。what 情報は状態が変化することに報告すればよいが、idle time は以前の情報が必要である。

以上のことから、idle time 情報だけを information server のデータベースに保存しておく必要がある。そして、ある一定時間経過したものを差分として報告するか、もしくは、定期的に user server に報告すれば良いことになる。

## 第 6 章

### domain server

#### 6.1 domain server の役割と機能

本来、ユーザが単一のシステム内だけで作業を行なうのであれば、ユーザ情報管理システムでは user server と information server の 2 つのサーバで十分耐え得るが、この管理システムは、広域分散環境における発展性、拡張性を十分考慮したものでなければならぬので、この domain server が必要になる。広域分散環境下では、1 人のユーザに対して 2 つ以上のアカウントが考えられる。例えば、koichi なるユーザが、

```
/jp/ac/u-tokyo/koichi  
/jp/ac/keio/koichi
```

といった 2 つのアカウントを持つことが考えられる。このように 1 人のユーザが複数のシステムに跨ってアカウントを持つと、そのユーザの情報は分散してしまう。先の例では、u-tokyo のシステム内での情報と、keio のシステム内での情報の 2 つが同時に存在しかねない。従って情報の信頼性がなくなってしまう。このことから情報はいくらユーザがアカウントを持っていようとも、ユーザごとに 1 ヶ所にまとめて保存することが望ましい。

こうしたことから、domain server は、情報を取り込むにしても、提供するにしても、その対象となるユーザの情報を管理する user server の位置をクライアントに提供する。言い換えれば、information server に user server のアクセスポイントを返したり、クライアントプロセスに、対象である user の情報を管理する (ユーザの本名の属するシステム内で起動している) user server のアクセスポイントを返したりする。

