

DHT を利用した地図分散型コンテンツ共有システム

北 望[†] 重野 寛[‡]

慶應義塾大学大学院理工学研究科[†] 慶應義塾大学理工学部[‡]

本稿では、DHT を用いて分散したコンテンツを地図上で視覚的に共有できる、地図分散型コンテンツ共有システムを提案する。提案システムでは、地図上に画像や口コミ情報などのコンテンツを登録することで、場所に関連したコンテンツをわかりやすく取得することが可能となる。また地図自体も分散することでスケーラビリティを高めている。視覚的なコンテンツ取得を可能とするため、コンテンツ同士の関連をメタデータに記述し、地図間の関連付けと地図とコンテンツ間の関連付けを行っている。さらに、提案システムのプロトタイプシステムを実装し、その評価を行った。

A contents gathering system using distributed maps on DHT-based P2P network

Nozomu KITA[†] Hiroshi SHIGENO[‡]

Faculty of Science and Technology, Keio University[†]
Graduate School of Science and Technology, Keio University[‡]

In this paper, we propose a DHT-based contents sharing system where maps are also distributed to visually share contents. Using the proposed system, the user can register and acquire contents such as images and even word-of-mouth information that are related to a certain place on the map with ease. In addition, we distribute the maps itself to enhance scalability. In the proposed system, the relationship between contents are described in metadata format to link multiple maps and to also link the contents to maps. This enables visual contents acquisition. We furthermore implemented a prototype system of the proposed system and carried out performance evaluations.

1 はじめに

これまで、画像やテキストなどのコンテンツをユーザ間で共有するシステムについては数多くの研究が行われてきた [1]。例えば、Web 上で多数の人が情報を書き込むことで情報が蓄えられる Wiki はコンテンツ共有システムの一つである。しかしながら、Web 上のシステムのほとんどはサーバに情報を蓄えるものであるため、サーバに障害が起こった場合にシステムが完全に停止してしまうという耐障害性の問題が存在する。また、蓄えられるデータ量がサーバの容量に依存してしまう他、アクセスが集中すると処理量が多くなってしまうというスケーラビリティの問題もある。

これらの問題を解決する手段として登場したのが P2P コンテンツ共有である。P2P ネットワークでは、接続された対等な存在であるピア同士が資源やサービスを直接やりとりする。そのため、全てのコン

ピュータが同等に機能してコンテンツはピア間で分散されるため、サーバの存在による耐障害性とスケーラビリティの問題を解決することが可能となった。

P2P ネットワークにおけるコンテンツ共有アプリケーションで有名なものとして Napstar[2] と Gnutella[3] がある。これらのアプリケーションは、クエリルーティングやインデックスの配置方式で様々な改良がなされているが、検索方式はファイル名とディレクトリ名のキーワードマッチングのみであった。個人が所有するコンテンツはテキストのみならず、画像を中心として多種のマルチメディアコンテンツを含むと考えられるが、キーワードマッチングのみを用いる環境ではユーザにとって使いやすくない。そのため、確実なキーワードが関連付けられたコンテンツの検索にしか適さない、という問題があった。また、P2P ネットワークが大規模になればなるほど、遠方のノードが持つコンテンツの取得に時間がかかるという問

題も生じた。

そこで、キーワードマッチングではなく視覚的にコンテンツを共有するため、地図上で画像やロコミ情報などの場所に関連付けられたコンテンツを共有するシステムを考案した。本稿では、DHT を用いて分散したコンテンツを地図上で視覚的に共有できる、地図分散型コンテンツ共有システムを提案する。

提案システムでは DHT を利用することで、ネットワークが大規模になってもコンテンツ探索にかかる時間を抑えられる。また、地図上でコンテンツ共有を行うため、キーワードマッチングでは検索しづらい画像などのコンテンツを視覚的にわかりやすく取得することが可能となる。この際、地図がコンテンツ共有のための土台となるので、利用する地図自体も他のコンテンツと同様に DHT で分散させて特定のノードへの負荷がかからないようにする。

分散した地図とコンテンツの関連付けは、メタデータ内に探索キーを保存することで行っており、この関連付けにより視覚的なコンテンツ取得が可能となっている。さらに、提案システムのプロトタイプシステムを実装し、その評価を行った。

2 関連研究

本研究に関連している研究として、まず MapWiki[1] が挙げられる。これは Wiki システムを共有地図上で実現し、ユビキタス環境におけるロコミ型コンテンツ流通を実現するという研究であり、コンセプトとして実フィールドとバーチャル環境の間の双方向のコラボレーションを促進し、ユビキタス環境を有効活用したコンテンツ流通の実現を目指している。

実装例として、Google マップ API を利用した GMapWiki を開発しており、GoogleMap 上で Wiki コンテンツを流通できるようになっている。サーバ／クライアント間では RDF を利用して Wiki データを送受信できるようになっている。

提案システムでは Google マップといった Web 上の地図を利用するのではなく、ユーザ自身が用意した地図を利用することで、サーバという障害点を持たない、スケーラビリティの高いコンテンツ共有環境を実現する。

また、DHT を利用した研究例として、P2P データ共有技術を活用した Web コンテンツの分散管理[4]がある。これは DHT に関する研究事例を活用し、Web コンテンツの分散管理システムを提案、その実

現方法や解決すべき技術的課題について議論を行っている研究である。具体的には、MIT で開発された Cooperative File System(CFS)[5]の実装をベースに、OceanStore[6]と呼ばれるインフラストラクチャを用いた Web コンテンツ分散管理システムを提案している。

本稿ではシステムの提案に終始せず、プロトタイプシステムの実装まで行ってその実用性を確認する。

3 システムの提案

3.1 DHT を利用した地図分散型コンテンツ共有システム

DHT を用いて構築した P2P ネットワーク上にコンテンツを分散させ、地図上で共有することを可能とするシステムを提案する。DHT を利用することにより、ネットワークが大規模になってもコンテンツ探索にかかる時間を抑えられるようになる。地図は各ノードに分散し、特定のノードに負荷が集中することがないようにしている。また実装画面に地図を表示し、その地図上にコンテンツを表示することで、視覚的にわかりやすいコンテンツ共有を実現している。

3.2 提案システムの概要

提案システムは DHT を用いて各ノードに分散させた地図上に、ユーザ自身が自分の好きなコンテンツを登録するユーザ参加型 P2P システムである。どのようなデータが登録されるかあらかじめ予測できないことや、ユーザが自身で登録を行わないとコンテンツが増えていかないうことを考慮し、多種のデータを登録できる高い汎用性を持たせた。図1にシステムの概要図を示す。図下部は DHT による P2P ネットワークでノードが複数個接続されている様子を表しており、各ノードが分散された地図をコンテンツとして保持している。また図には示していないが、各ノードは画像などのコンテンツも保持している。図上部はそのネットワーク上で分散地図を用いてコンテンツ共有を行っている様子を示している。分散地図同士は上位地図と下位地図として関連付けがなされており、また地図と地図上に登録されたコンテンツも関連付けられている。ここで上位地図はある地図と比較して縮尺が小さく、広い地域を表現する地図を指し、下位地図は縮尺が大きく、狭い地域を表現する地図を指す。

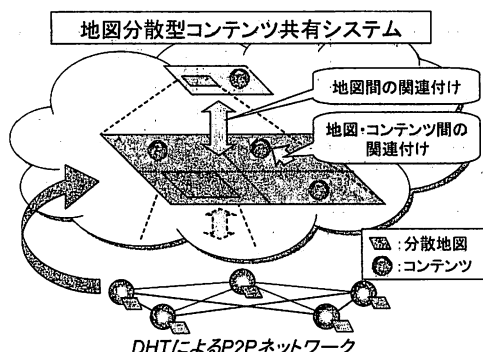


図 1: システム概要図

3.3 構成要素

3.3.1 地図上でのコンテンツ共有

個人が所有するコンテンツはテキストに限らず、画像を中心としたマルチメディアコンテンツを含むと考えられるが、キーワードマッチングのみを用いる環境ではユーザにとって使いやすいインデックスを作成できるとは考えにくい。そこでキーワードマッチングではなく、地図上で画像などの場所に関連付けられたコンテンツを共有することを考えた。具体的には、地図をコンテンツ取得のプラットフォームとし、システムで地図を表示させ、その地図上にコンテンツをアイコン表示する。地図上のアイコンをクリックすることでコンテンツのキーを取得でき、またコンテンツの登録も地図上で座標指定することで行う。

しかし、地図上でコンテンツの取得を行う際に考えられる問題として、地図を表示する際に毎度毎度地図上に登録されているコンテンツを全て取得していると、ネットワーク負荷が高くなってしまうという問題がある。これは、例えば地図上に100KBの地図が100枚登録されている場合では地図を見るだけで10MBものデータを取得しなくてはならないことを指す。また、直接コンテンツを取得すると、そのコンテンツの種類の判別ができず、アプリケーション側で種類に応じて処理を変えたい場合に対応できないという問題も存在する。ここでコンテンツの種類として、地図・画像・映像・ロコミ情報といったテキストなどを想定している。また処理例として、画像であれば画面に表示し、映像であればメディアプレイヤー起動する、などを想定している。

これらの課題を解決する方法として、提案システ

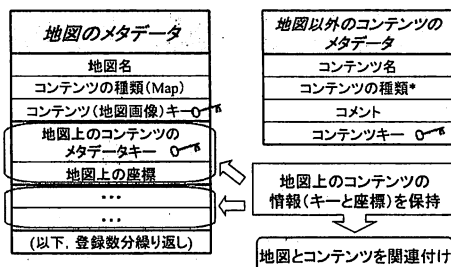


図 2: メタデータの比較

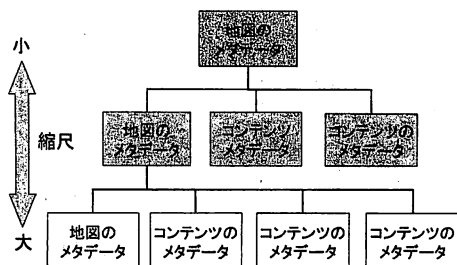
ムではコンテンツに対するメタデータを用意することとした。コンテンツを取得する代わりにメタデータを取得することで、地図表示時の負荷を軽減でき、さらにメタデータ内にコンテンツの種類を記述しておくことで、コンテンツの取得前にコンテンツに対する処理を決定できるようになる。

3.3.2 メタデータによる関連付け

提案システムでは、DHT登録時にコンテンツのメタデータを作成する。メタデータには構造が異なる2種類が存在し、「地図コンテンツのメタデータ」と「地図以外のコンテンツのメタデータ」に分類する。図2に2種類のメタデータの内容を示す。

2種類のメタデータは、共にコンテンツ名、種類、コンテンツの探索キーをフィールドとして持つ。このように、メタデータ内にコンテンツキーを保持することで、コンテンツとメタデータの関連付けが行われている。作成されたメタデータはコンテンツと同様にDHTに登録され、その際生成されるメタデータキーはコンテンツの登録を行った地図のメタデータに関連付けられる。地図のメタデータ特有のフィールドとして、地図上に登録されたコンテンツの情報を保持するフィールドが存在する。これらの情報は登録されているコンテンツそれぞれに対して一組ずつ保存される。したがって、地図のメタデータを取得できれば地図上のコンテンツの情報も取得できるようになり、地図とコンテンツの関連付けが行われている。

ここで、地図上には画像などのコンテンツの他、縮尺の異なる地図も登録される。そのため、地図のメタデータ内に縮尺の異なる地図のメタデータが保持され、さらにその地図には他の地図が登録される...というように、地図は縮尺による階層構造を持つことになる(図3)。



注：図中の「コンテンツ」は
地図以外の全てのコンテンツを指す

図 3: 地図の縮尺による階層構造

3.3.3 DHT へのコンテンツ登録と取得

DHT へのコンテンツの登録は、地図上で座標を指定して行う。登録の様子を図 4 に示す。コンテンツを DHT に登録するとコンテンツの探索キーが生成され、その生成されたコンテンツキーはコンテンツのメタデータ内に保存される。そしてこのメタデータもコンテンツに続いて DHT に登録されるが、その際に今度はメタデータの探索キーが生成され、そのキーはコンテンツに登録した地図のメタデータ内に保存される。最後に地図のメタデータを更新することで、DHT への登録が完了する。

メタデータの更新は、まず存在するメタデータを削除した後、すぐに新しいメタデータを同一のハッシュ値（つまり同一ファイル名）で登録し直すことで行われる。これは、メタデータは他のメタデータから関連付けられ参照されているので、同一のハッシュ名を維持する必要があるためである。削除はメタデータのキーで探索を行い、みつかったノードに削除要求を出すことで行われる。そのため、削除を行ってから登録が完了するまでの時間、更新対象のメタデータへはアクセスができなくなるという問題が残されるが、メタデータの更新を確実にを行うためにこのような仕様となっている。

DHT からコンテンツを取得する様子を図 5 に示す。コンテンツは地図のメタデータに関連付けられているので、最初に地図のメタデータを取得する。地図のメタデータの取得は、システム起動時と地図上の地図のアイコンをクリックするなど、ユーザからの明示的な指示により行われる。地図のメタデータ取得後、まずメタデータ内のコンテンツキーからコンテンツである地図画像を取得する。その後、地図上に登録されたコンテンツのメタデータを順次探

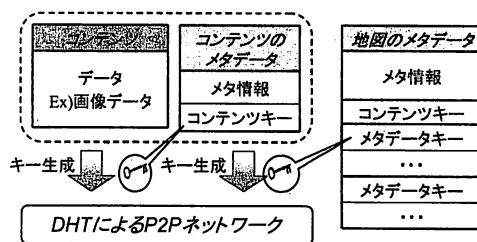


図 4: DHT へのコンテンツ登録

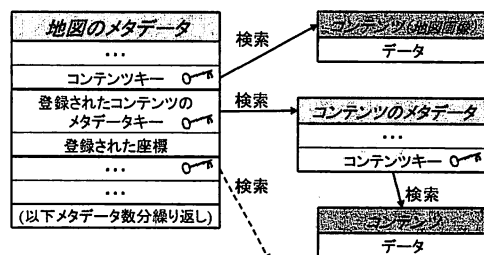


図 5: DHT からのコンテンツ取得

索し、コンテンツのメタデータを取得していく。こうして取得されたメタデータは地図上にアイコンで表示される。ユーザが地図上に表示されたアイコンをクリックするたびに、取得済みのコンテンツのメタデータからコンテンツキーを用いて DHT での探索を行い、コンテンツ本体を取得する。

4 プロトタイプシステムの実装

4.1 実装環境

提案システムのプロトタイプシステムの実装にあたっては、各接続 PC 上で提案システムを 10 個ずつ立ち上げ、PC1 台に付き仮想的に 10 個のノードを立ち上げることで P2P ネットワークを構築した。この手法を用いることで、PC10 台で最大 100 個の仮想ノードによる DHT ネットワークを構築することが可能となる。

また、プロトタイプシステムの実装には Java を利用している。これは、システムをどのようなプラットフォームであっても相互に利用可能なものとするためである。

4.2 システム構成と特徴

システムの構成を図 6 に示す。図の下側はネットワーク部で、FreePastry で構築されている。その上

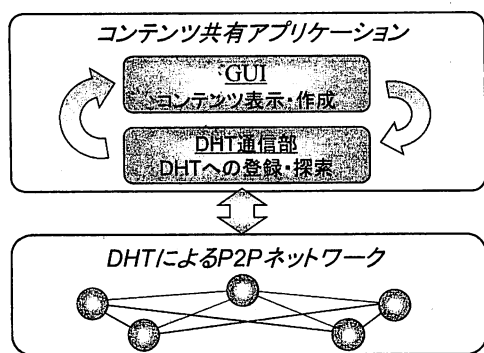


図 6: システム構成

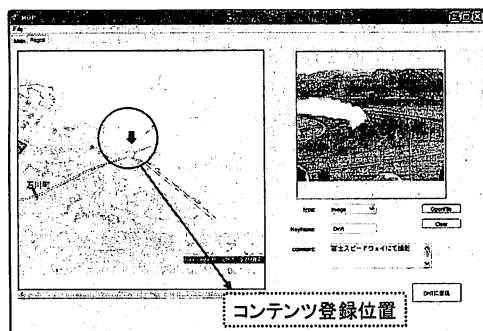


図 7: コンテンツの登録方法

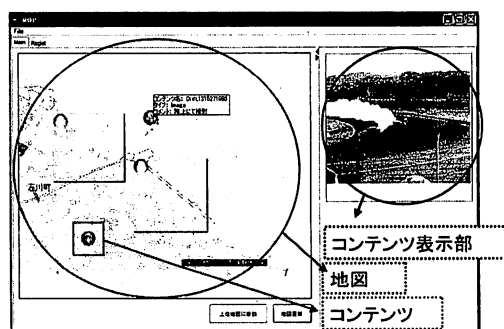


図 8: コンテンツ取得画面

にアプリケーション部として、地図上でのコンテンツ共有を実現するアプリケーションを乗せる構成となっている。

実装はアプリケーション部とネットワーク部とを分けて行った。ネットワーク部はBSDライセンスのソフトウェアであるFreePastry[7]を利用した。このFreePastryはDHTの一種であるPastryをJavaで実装したものである。Pastryを利用しているため、ノード数が多くなっても高速な探索・ルーティングが可能となる。またPureP2P型であるため、サーバに依存することがなく、耐障害性・スケーラビリティに優れたシステムを構築できる。

4.3 実行例

4.3.1 コンテンツの登録

コンテンツの登録の実行例について詳しく説明する。コンテンツ登録は図7の登録画面を利用して行う。この図は地図以外のコンテンツ（ここではIMAGEタイプ）を登録するものとする。

まずOpenFileボタンで登録するコンテンツを選択した後、画面右でコンテンツの名称、コンテンツタイプ、コンテンツのコメントを設定する。その後、画面左の地図上でコンテンツを登録したい場所を選択し、「DHTに登録」ボタンを押すことで登録が完了する。

また、地図コンテンツを登録する場合は、先ほどと同様にコンテンツの名称、コンテンツタイプを設定し、画面左の地図上でマウスをドラッグして地図の登録範囲を指定する。

4.3.2 コンテンツの取得

コンテンツ取得を行うためのメイン画面を図8に示す。メイン画面の左側には大きく地図が表示され、この上に登録されたコンテンツがアイコンで表示されている。このアイコンにマウスを重ねることで、コンテンツの情報がツールチップ表示され、アイコンをクリックすることでコンテンツを取得することができる。ここでコンテンツの種類が画像であった場合は、画面右のコンテンツ表示部に画像が表示される。

5 評価

5.1 実験環境

プロトタイプシステムの実用性を評価するため、コンテンツの探索時間の測定とその評価を行った。

表 1: 実験で使用するパラメータ

ノード数 V	10, 20, 50, 100
コンテンツ数 N_c	10, 20, 50, 100, 200, 500, 1000
探索回数 S_c	10
レプリケーション数 R	2

本実験では1台のPCにつき仮想的に10個のノードを立ち上げ、それらのノードでDHTネットワークを構築して実験を行った。主なパラメータを表1に示す。実験は表に示した接続ノード数 V とコンテンツ数 N_c の全ての組み合わせで行った。また、システム中にいくつのレプリカを持たせるかを決定するレプリケーション数 R はFreePastryの初期値である2を採用した。

以下、実験の手順を説明する。(1) あらかじめ、ある1枚の地図上にコンテンツをコンテンツ数 N_c だけ登録する。(2) 登録した地図のメタデータを取得し、そのメタデータから地図上のコンテンツのメタデータを全て取得するまでの探索時間を計測する。(3) この探索は S_c 回繰り返して、その平均時間を評価する。

5.2 結果と考察

本実験より得られた結果を図9にまとめる。結果のグラフは横軸がコンテンツ数で単位は個数、縦軸が探索時間で単位は秒である。

図9に示すように、コンテンツ数が増加するにつれて探索時間は比例して増加している。これはコンテンツ1つ1つの探索はノード数に対してlogオーダーで済むが、本実験ではコンテンツを1つ1つ繰り返し探索していることより、コンテンツ数が増えるに従い、ノード数とは関係なく比例して探索時間が増えるためである。

また、ノード数が10から1000まで増えても探索時間はほとんど変わっていない。これはノード数が増えてもコンテンツ1つの探索にかかる時間が増加していないことを示している。探索時間は大きい値を示しているが、ノード数が100でコンテンツが1000個の場合、コンテンツ1つあたりでは0.167秒で取得できており十分小さい値を取っている。さらに、グラフの傾きが直線を維持していることから、この条件の中においてはシステムに過剰な負荷がかかることはなかった。

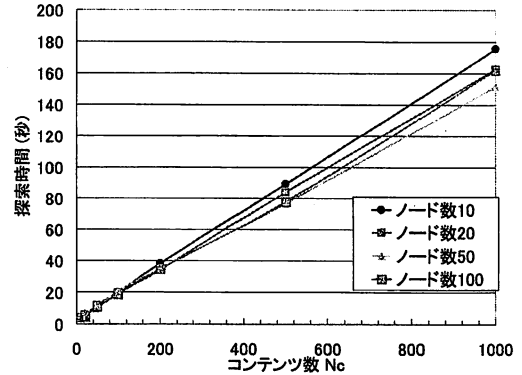


図 9: コンテンツ探索時間

6 おわりに

本稿では、DHTを用いて構築したP2Pネットワーク上において、コンテンツを地図上で視覚的に共有できるシステムを提案し、そのプロトタイプシステムの実装・評価を行った。評価実験において、ノード数とコンテンツ数を増やした場合でもシステムに過剰な負荷がかかることはなかったため、提案システムの実用性を示すことができたと考える。

参考文献

- [1] 寺西裕一, 鎌原淳三, 下條真司. Mapwiki: 共有地図を用いたユビキタスコンテンツ流通環境. 情報処理学会 DPS ワークショップ論文集, pp. 414-419, 2005.
- [2] Napster. <http://www.napster.com/>.
- [3] gnutella. <http://www.gnutella.com/>.
- [4] 藤田昭人, 石橋勇人, 大西克実, 中野秀男. P2pデータ共有技術を活用したwebコンテンツの分散管理. 情報処理学会 DPS ワークショップ論文集, Vol. 2005, pp. 437-441, 2005.
- [5] Dabed F. et al. Widearea cooperative storage with cfs. *Proceedings of the Symposium on Operating Systems Principles(SOSP)*, 10 2001.
- [6] kubiawicz J et al. Oceanstore: An architecture for global-scale persistent storage. *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems(ASPLOS)*, 12 2000.
- [7] Freepastry. <http://freepastry.org/>.