協調サーチエンジンによるクラウド検索

# 1 はじめに

近年、クラウドと言う言葉をよく聞くようになった。クラウドとは、インターネットを経由して必要な分のリソースを借りるサービスである。代表的なもので、AmazonWebServicesやGoogleAppEngineなどが挙げられる。また特徴は、負荷に応じて簡単、または自動にサーバの台数を増やすことができる。よって、負荷を予測して物理サーバを用意する必要が無く、無駄な費用を抑えることができる。そのため、企業においてもクラウドを積極的に活用しようという機運がある。だが、問題となるのがITガバナンスだ。そもそもGoogleAppEngineなどの代表的なクラウドサービスは、パブリッククラウドと呼ばれる。パブリッククラウドとは、さまざまな企業が利用し、データの保存場所は把握できなく、またトラブル時には原因追求が困難になる可能性がある。よって、企業が持つ機密情報を、第三者が提供するクラウドに保存してよいのかといった問題が生じる。こうしたことから、プライベートクラウドに注目を集めている。パブリッククラウドでITガバナンスを効かせることはできない。しかし、独自に構築するプライベートクラウドであれば、ITガバナンスを効かせることができる。よって、パブリッククラウドと平行してプライベートクラウドも活用されると考えられる。

クラウドの活用が進むと、クラウドをオンラインストレージとして利用することが予測される。そのため、企業や個人が生み出した数多くのデータが集約される。そのデータを効率良く利用するためにも、検索技術は欠かせないものである。よって、アップロードされた直後から検索できなければ、データの活用とは言えない。また、クラウドの環境が数多くのサーバで構成されている。よって、分散検索ができるシステム設計である必要がある。よって、本研究ではこの二点に対して、研究を行う。

以下、本文の構成である。まず、第2章では、クラウドで分散検索を行うためのシステムとして協調サーチエンジンと、クラウドのデータベースで用いられているKey-Valueストアについて説明をする。第3章では、クラウド検索のシステム設計を説明し、最後に今後の課題を述べる。

# 2 関連研究

2.1　協調サーチエンジン

協調サーチエンジンは、本研究室で研究されていた分散型のサーチエンジンである。協調サーチエンジンの構成は、ユーザからクエリを受け付けるのがLocal Meta Search Engine (LMSE)である。このサーバはまず、どこに探しているデータが、あるのかを知るために、Location Server (LS)に問い合わせる。LSは、クエリに対しアクセス先のサーバアドレスを管理している。それにより、検索を依頼するサーバを決定し、そのアクセス先に対して検索要求を各LMSEに送信する。LMSE内にあるLocal Search Engine (LSE)が保持しているインデックに対して検索を行い、結果を返す。ユーザからクエリを受けたLMSEはそれらの結果をまとめて、ユーザに返すといった、システムの構成となっている。その構成図が図1である。

本研究では、協調サーチエンジンのLSのように分散している検索サーバに対して、アクセス先を決定する。この構造は、クラウド検索に対して効率よく検索が可能であると考える。

図1　協調サーチエンジン

2.2　Key-Valueストア

データベースといえば、Relational Database(RDB)が使われてきた。だが、クラウドという複数のサーバで構成されている環境では、RDBではなくKey-Valueストアと呼ばれるデータベースが使われるようになった。特徴としては、Keyに対してValueを保持する構造である(図2)。シンプルな構造によって、高速性が生まれ、また分散もしやすくなる。

本研究では、このKey-Valueストアを用いて、協調サーチエンジンでいうLSのデータベースに使用する。

図2　Key-Valueストアの構造

# 4 設計

クラウドは、複数のサーバから構成されている。よって、簡単に分散処理できる設計が重要となる。

システムの構成としては、図3である。ユーザは、フロントにアクセスし、ユーザ確認をする。それによりユーザの取得できる情報の範囲を決定する。その後、クエリの内容から検索対象のサーバを探すために、Metaサーバにアクセスし、アクセス先情報を得る。あとはオープンソースであるApache Solrの分散検索機能を利用し、分散検索をする。分散検索された結果は依頼したApache Solrにまとめられる。だが、ランキングは、各Apache Solr内でランキングされた結果をまとめるだけである。なので、最終的に返ってくるランキングは正しいランキングでは無い、よって、ユーザに返す前にランキングを直す処理をする。文書の収集では、複数のクローラが並列的に行い、処理を分散させる。それにより、アップロード直後に検索できるようにする。文書の対象は、HTML、PDF、Microsoft Officeのファイルデータを予定している。その構成図が図3である。

図3　クラウド検索

# 5 今後の課題

実装を行い、分散処理ができるかの判断や、処理速度の問題などを検証する。クラウドにあったセキュリティを考える。

# 謝辞

研究を指導してくださり、適切な助言を与えてくれた教員および大学院生等に感謝の意を表する。

### 参考文献

1. A.Sato, T.Sato and T.Toyo; How to write papers, J. ACM, 25, 10, (1990).
2. K.Ito and M.Joho; The art of programming, IEEE Trans. computer, 30, 4, (1995).
3. 城宝; 卒業研究マニュアル, 東洋出版, (1996).

### 付録

ここではその他の注意事項について述べる。

* 論文は「です・ます」調ではなく「だ・である」調の文体で書く。
* 専門用語は統一して使用し､解説を加えること｡
* 参考文献は参照すること。参照していない文献は不要である。
* 英文、数字は半角文字で書く。全角は正しく整形されないため使用禁止。
* 英語にはスペルチェックを行う。
* 半角カタカナは使用禁止。
* 要旨の書式は仕様を厳守すること。要求された書式を達成できずに妥協することは許されない。
* 枚数は１件につき２枚（両面時２頁）に制限する。
* 上下左右の余白を2cm以上開けること。
* 表題は、12ポイント、ゴシック体とし、中央に揃えること。また、題目と氏名行の間と本文の間を1行空けること。副題は、9ポイント以下の明朝体とする。
* その他の文字は、10ポイント明朝体とすること。ただし、各項目（「1. はじめに」など）は、ゴシック体とすること。また、参考文献及び付録は9ポイントとすること。
* 各項目は上記のように左右２段組とすること。間隔は10ポイント２文字相当（約7.5mm）とする。
* 各章節の区切りは1行以上空けること。
* 図と表の説明文は 「Fig. 1. Experimental apparatus.」、 「Table 1. Physical constants of materials.」のように書くこと。なお、図では下に、表では上につけること。
* なるべくLaTeXまたはMS Wordで書くことを推奨する｡LaTeXまたはMS Word 97用のスタイルファイルが用意されている｡
* 発表と内容が一致していること｡
* 研究を引き継いでいる場合には前任者との相違を明確にすること｡