

Web アプリケーション分野の技術動向と展望

教授 小山 裕司

2009年12月のInfoTalk#13では、今後のIT動向として、クラウド、ソーシャルグラフ等と一緒にNoSQLを取り上げた。あれから約2年が経過したが、今回もWebアプリケーション分野の技術動向と展望として、NoSQLの台頭を取り上げたい。

《RDB》

DB（データベース）は、データを集めて管理し、検索と抽出等の再利用ができるようにするためのものである。これらを高速・安定・容易に実現するために各種の工夫が行われている。DBの黎明期に相当する1970年台頃にはDBをどのように実現するか（データをどのように配置するか等のいわゆるデータモデル）によって複数の種類があったが、現在ではRDB（関係データベース）が圧倒的に広く普及している（表1）。RDBは、データの操作には関係代数あるいは関係論理を使い、これらを表現する言語としてSQLが設計され、普及している（1986年頃から標準規格が存在する）。RDB以外のDBとしては、Unix DBMの一種であるBerkeley DB、GNU gdbm等があるが、存在感は薄い。

表 1: RDB (SQL の実装)

Oracle Database	DB2	Sybase ASE
Microsoft SQL Server	MySQL	PostgreSQL
SQLite	FireBird	

《Web アプリケーション》

2000年ごろからのWebアプリケーションの普及に乗じて、RDBも急速に普及した（図1）。RDBは、Webアプリケーションが取り扱う各種のデータ（売上情報、利用者のアカウント情報等）を蓄積する目的で使われている。

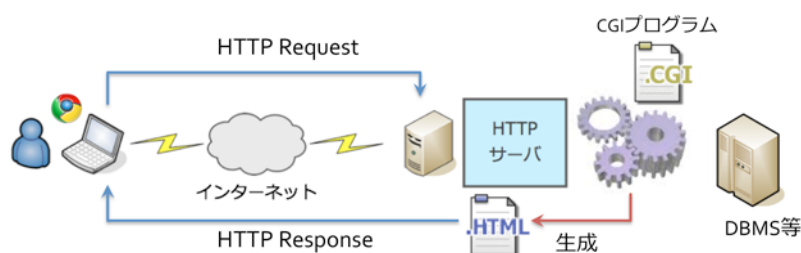


図 1: Web アプリケーションの仕組み

Webアプリケーションと、通常のPC上のWord、Excel等のアプリケーションの違いに連続稼働時間（寿命）がある。最近のOSはマルチタスクが一般的で、またPCの搭載物理メモリも数GBであるため、PCアプリケーションのプログラムの寿命は数時間から場合によっては数日に伸びる傾向にある。PCアプリケーションは起動時にファイルからデータを読み込み、編集作業等を行い、作業終了時に編集済み

のデータをファイルに保存する。念のため作業の途中で保存することもあるが、原則、編集中のデータはプログラムの実行中のみ存在し、プログラムが終了すると破棄される揮発オブジェクト (volatile object) によって保持される。C 言語等の変数を想像していただきたい。

これに対して Web アプリケーションで使われる CGI プログラムの寿命は概して 1 秒未満である。CGI アプリケーションは HTTP リクエストを受けて起動し、動的に HTML コンテンツを生成するが (図 1)、これが数秒かかってしまうようであれば利用者にストレスを与えてしまう。このように寿命が短く頻繁に起動されるプログラムでは、理想的にはプログラムの寿命とは独立し、オブジェクトとして存在し続ける永続オブジェクト (persistent object) が必要とされる。永続オブジェクト自体は 1990 年ごろ積極的に研究されていたが、CGI プログラムの特性から再度息を吹き返した。永続オブジェクトはプログラムだけでは実現出来無いので、いちいち手動で行うか、O/R マッパー等を利用するかは別にしても、データの受け皿としては外部の記憶装置 (ファイル、DB 等) に頼る必要がある。永続オブジェクトの実現に、RDB レベルの機能が必要だというわけでは無いが、相当の数の RDB が Web アプリケーションの永続オブジェクトの実現相当にも使われている。

《NoSQL の台頭》

インターネット及び Web アプリケーションが普及し、非常に多数のリクエストを処理する必要があると、HTTP サーバ及びアプリケーションサーバは台数増によるスケールアウトが可能であるが、RDB はスケールアウトが難しいため、RDB というボトルネックをどのように解決するかという問題が生じる。

分散システムでは、Consistency (一貫性)、Availability (可用性)、Partition Tolerance (分割耐性) の 3 要素のうち 2 要素しか満足することが出来無いという定理である (CAP 定理[1,2])。RDB は一貫性と可用性をできるだけ保証するかわりに分割耐性を犠牲にする CA 型であり、ネットワークが切れたり遅延したりした場合は動作が保証出来無い。

前節でも指摘しているように、Web アプリケーションの永続オブジェクトの実装等では、RDB レベルの機能がどうしても必要だというわけでは無いので、最近では RDB 以外の DB を使い、ボトルネックを減らす試みが流行っている。Facebook が使っている Cassandra、Google の BigTable、Zynga が使っている Couchbase (Membase) 等、約 40 種類が新規に開発され、日本でも複数の開発が行われている。これからの RDB 以外の新しい DB は NoSQL (Not Only SQL) という名称で総称される。NoSQL は概して、RDB ほど高機能では無いが、性能重視・台数増によるスケールアウト可という特徴があるが、ほかにもいろいろと特徴 (長所・短所) が存在する。これらは、CAP 定理の CA 型・AP 型・CP 型という分類のほかに、ドキュメント指向・カラム指向・KVS (Key Value Store) 等のデータモデルでも分類できる (図 2)。AWS の SimpleDB では、ドキュメント指向で、P と A が保証される AP 型で、C に対する弱点がある。また、BigTable、Hbase はカラム指向で、C と P が保証される CP 型で、A に対する弱点がある。

従来は、DB は RDB のみが選択肢で、表 1 の中から、商用か OSS かという選択をする程度だったものが、最近では NoSQL の開発及び活用が積極的に行われた結果、目的及び制約から DB の選択に相当の幅が出来ている。これはどの DB を選択するかが、情報システムのアーキテクチャ及びプロジェクトの成否を左右する可能性もあるということも意味するわけである。NoSQL はまだ積極的に研究開発が行われている最中であるから、気軽に活用できるというものでは無いが、次第に障壁は取り除かれ、Amazon

SingleDB, Google App Engine のようにクラウド環境で提供されているものもあるので、機会があれば試したり、勉強したりしてみるのもいいだろう。

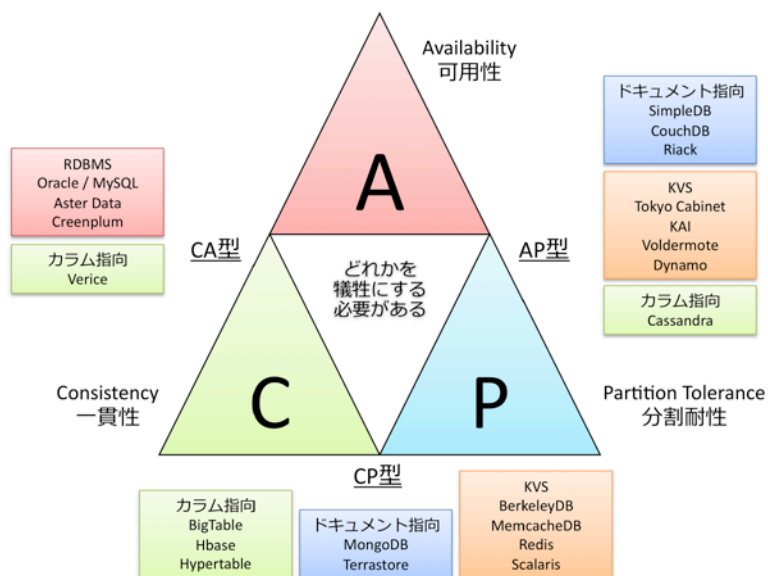


図 2: CAP 定理

- [1] Eric Brewer, “Towards Robust Distributed Systems”, Symposium on Principles of Distributed Computing, <http://www.cs.berkeley.edu/~brewer/cs262b-2004/PODC-keynote.pdf>, 2000
- [2] Nancy Lynch and Seth Gilbert, “Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services”, ACM SIGACT News, Volume 33 Issue 2 (2002), pg. 51-59.
- [3] Nathan Hurst, “Visual Guide to NoSQL Systems”, <http://blog.nahurst.com/visual-guide-to-nosql-systems>, March 15, 2010