

Bidirectional Collaborative Data Sharing の概要

English follows Japanese

武市正人 2019/12/29

Collaborative Data Sharing (協調型データ共有システム)

Collaborative Data Sharing (CDS, 協調型データ共有) システムは、固有のデータを持ち他のサイトと一部のデータを共有しつつそれぞれが特定のアプリケーションを実行する複数のサイトからなる。ネットワークで結合されたサイトは互いに独立して実行されるが、共有データの更新は協働して行われる。すなわち、共有データに関しては、あるサイトにおける更新はそれ以外のサイトにおいても尊重され、個々のサイトの方針に基づいて保有するデータに反映される。

データ共有のシステムとしては、もっぱらデータを提供するサーバとそのデータを受け取って処理を行うクライアントからなるサーバ・クライアント方式も考えられるが、CDSシステムではより一般的な協調を実現する Peer-to-Peer (P2P) による構成を目指している。

Bidirectional Collaborative Data Sharing

Bidirectional Collaborative Data Sharing (BCDS) は CDS を実現するP2Pアーキテクチャの一つであり、それぞれのサイトに固有のアプリケーションによる処理と共有データの授受の機能を分離することにより、アプリケーションの開発を容易にするとともに、共有データが更新されたときにシステム内のすべてのサイトにわたってその更新を反映させて共有データの整合性を確保している。BCDS システムにおけるサイトは BCDS Agent によって実現され、複数の BCDS Agent が相互に TCP/IP によるネットワーク通信で結合される。

“BCDS” が “CDS” に “Bidirectional” を冠する理由は、BCDS Agent によって共有されるデータの授受に際して、それぞれの BCDS Agent 内でデータの提供に関わる方針や共有データが他の BCDS Agent によって変更されたときの更新の方針などを双方向変換 (bidirectional transformation) によって実現していることによる。

BCDSシステムによる協調型データ共有システムの構成

協調型データ共有システムをBCDSによって構成する際には、一般のCDSシステムがBCDS Agentによって構成できることから、システムの総体の構成としては特段の配慮をしないといえるが、BCDS Agentの特徴を考慮することも有用であろう。

BCDSによるシステムの設計にあたっては、とくに以下の点に留意するのがよい(p.5の図を参照)。

- システムを構成するサイトごとのアプリケーションと対象とするデータの仕様を定める。他のサイトと共有するデータがある場合には、自らが維持するデータベースの項目に共有データの項目が付加されたSQLテーブルを対象としたアプリケーションをSQLクエリによって記述することを想定する。SQLクエリによる処理に相当する以外の言語による記述もありうるであろう。
- 他のサイトと共有するデータについて、そのデータの提供を求める(複数の)サイトの保有するデータ項目とそれらの中から求めるべきデータ項目を定める。共有データは、その提供元でSQLクエリ `SELECT c_1, c_2, ... FROM * WHERE ...` によって選択されるデータを想定すればよい。さらに、データ項目 `c_1, c_2, ...` ごとにそのデータ項目のデータを変更した際にその結果を提供元に返すかどうかを指定する。
- BCDSではサイトごとに他のサイトからデータの提供の要求を受け付けることになっている。自らが保有するデータのうちで他のサイトに提供することのできるデータ項目を定める。他のサイトと共有する個々のデータ項目について、提供先でのデータの変更結果を返すように求める(ReadWriteモード)か、提供先でのデータの変更にはこだわらない(ReadOnlyモード)かを指定する。さらに、ReadWriteモードで提供したデータ項目のデータが変更されたときに自ら保有するデータの更新として反映させるときの方針(update policy)を定める。標準的には、(複数の)提供先サイトから返されるデータの変更結果を先着順に自ら保有されるデータに反映させることになっている。

BCDS Agent の構成

BCDSシステムは、それぞれが固有のデータベースを維持しつつ外部との連携によってデータの共有を図る複数の BCDS Agent からなる。BCDSシステムには、複数の BCDS Agent によって共有されるデータをいずれかの BCDS Agent が更新したときに、関連するすべての BCDS Agent の保有するデータを自動的に同期して整合性を保つ仕組みが備わっている。

BCDS Agent は、自らが保有するデータを対象としたアプリケーションに加えて、他の BCDS Agent の要求に応じて自らが保有するデータ（の一部）を提供するサーバ機能と、共有したいデータを他の BCDS Agent に要求するクライアント機能を有する。サーバ機能とクライアント機能はアプリケーションとは独立して併行に実行される。ここで、自らが保有するデータには、BCDS Agent 固有のデータベース内のデータのほか、クライアント機能によって他の（複数の）BCDS Agent から提供を受ける共有データが含まれる。これらのデータは SQL データベースの表 (table) の形式で表現され、共有するデータ項目はこの表の列 (column) に対応している。

BCDS Agent が保有するデータが更新されるのは、固有のアプリケーションによる場合のほかに、クライアント機能によって他の（複数の）BCDS Agent から提供を受ける共有データが提供元で変更された場合、およびサーバ機能によって他の BCDS Agent に提供した共有データが提供先によって変更されてその結果が返されてきた場合がある。クライアント機能に関わる更新、サーバ機能に関わる更新は BCDS Agent のそれぞれの機能によってアプリケーションと併行して処理されるので、アプリケーションでそれを明示的に記述する必要はない。

BCDS Agent はサーバ機能によって自らが保有するデータを他の（複数の）BCDS Agent に共有データとして提供する。共有データ項目には、提供した共有データが提供先の BCDS Agent によって変更されたときにその結果が返されるもの (ReadWriteモードでの提供) と、提供先によって変更されてもその結果が返されないもの (ReadOnlyモードでの提供) とがある。

- ・ ReadWriteモードで提供された共有データが提供先で変更されて返されるデータは、BCDS Agent によって定められる更新方針(update policy)に基づいて自らが保有するデータの更新として反映させる。同一のデータ項目が複数の BCDS Agent に提供される場合には、提供先における変更のすべてが自らが保

有するデータの更新として反映されるとは限らない。場合によってはある提供先における変更結果が破棄される場合もありうる。このような状況が生じるのは、BCDSシステム内の BCDS Agent のそれぞれが相互に独立してデータを変更することを許容していることによる。複数の提供先からの共有データの変更が行われてそれらの間に競合が生じる場合の取扱いも BCDS Agent ごとの更新方針に依存する。標準的には、提供先からのデータの（有効な）変更が返されてきた時間的順序によるものとしている。

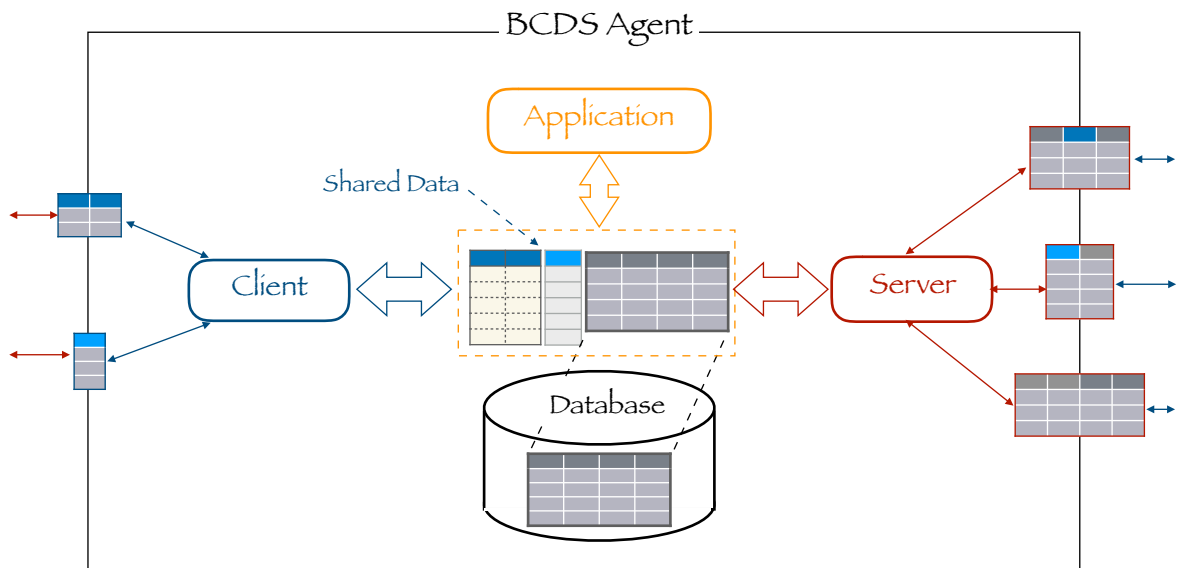
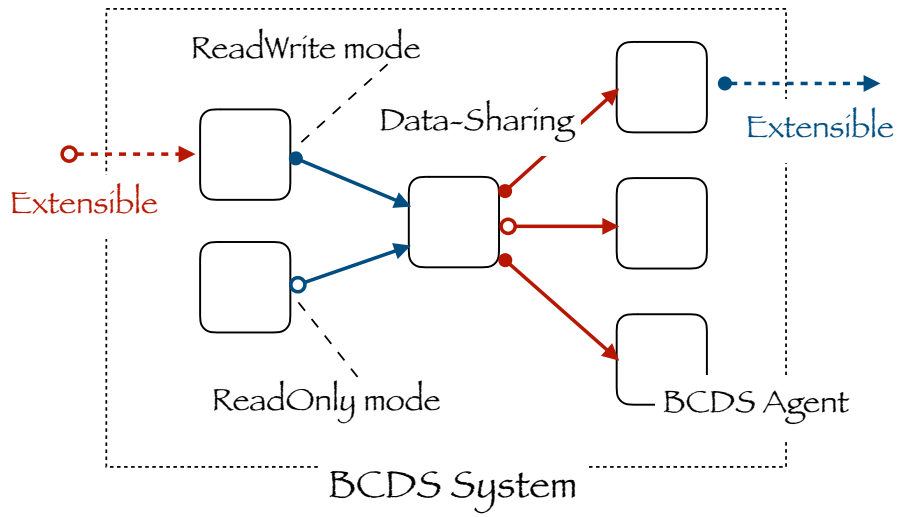
BCDS Agent のクライアント機能では、自らが求める他の（複数の）BCDS Agent が保有するデータをそれぞれの BCDS Agent のサーバ機能を通じて提供するように要求する。まず最初に、提供元の BCDS Agent に対して（複数の）データ項目を指定してデータの授受のための接続を確立する。この際、個々の項目についてReadWriteモードでの提供、あるいはReadOnlyモードでの提供のいずれであるのかを明示する。これが提供元の BCDS Agent の提供方針(service policy)と異なる際には接続が拒否される。接続が認められたときには、それ以降はこの接続を通してデータを受け取る。

- ・ BCDS Agent のクライアント機能は、自らが要求する共有データのために接続を確立する処理、提供元のサーバ機能により（提供元の保有するデータが更新されたことによって）送られてくるデータを受け取る処理、およびアプリケーションあるいはサーバ機能によって ReadWriteモードのデータ項目が変更されたときにそれを提供元に送る処理を実現している。これらの処理はアプリケーションとは併行して行われるのでアプリケーションで明示して記述する必要はない。

BCDS Agent のサーバ機能とクライアント機能によって構成される BCDSシステムは、一般的には Peer-to-Peer の接続によるグラフ構造をなす。BCDS Agent の固有のアプリケーションによるデータの更新はこの構造の上で自動的に伝播され、共有データのシステム上での整合性(Eventual Consistency)が保たれる。

- ・ BCDSシステムでは、随時に新たな BCDS Agent が追加されることもあり得るし、また、システム内の BCDS Agent は随時に停止したり、一部のクライアント機能を停止することも許容している。

Illustrative Figures



Overview of Bidirectional Collaborative Data Sharing

Masato Takeichi 2019/12/29

Collaborative Data Sharing

The Collaborative Data Sharing (CDS) system consists of multiple sites that have specific data and share some data with other sites, each running a specific application. The sites connected by the network are executed independently of each other, but the update of the shared data is performed in cooperation. That is, concerning shared data, updates at one site are respected at other sites and are reflected in data held based on the policies of each site.

As a data-sharing system, a server-client system consisting solely of a server that provides data and a client that receives and processes the data can be considered, but in a CDS system realizes more general Peer-to-Peer (P2P) cooperation.

Bidirectional Collaborative Data Sharing

Bidirectional Collaborative Data Sharing (BCDS) is one of the P2P architectures that realizes CDS. By separating the application that processes data and the functions for shared data between sites, the development of applications becomes facilitated. When shared data is updated, the updates are reflected across all sites in the system to ensure the integrity of the shared data. A site in the BCDS system is realized by a BCDS Agent, and multiple BCDS Agents are connected by the network communication using TCP / IP.

The reason that “BCDS” is named “Bidirectional” prefixed “CDS” is because the policy of providing and updating shared data at the time of execution is realized by the bidirectional transformation.

Configuration of the cooperative data-sharing system by BCDS system

When a collaborative data-sharing system is configured with BCDS, a general CDS system can be configured with BCDS Agent, and no special consideration is required for the overall configuration of the system. But it would also be useful to consider the characteristics of BCDS Agent.

In designing a system using BCDS, it is particularly important to pay attention to the following points.

- The application and the data for each site that constitutes the system should be specified. When there is data to be shared with other sites, it is assumed that an application is described by an SQL query that processes an SQL table in which the items of the shared data are added to the items of the database maintained by itself. There could be descriptions in languages other than the equivalent of processing by SQL queries.
- For data shared with other sites, determine the data items held by the provider sites and the data items to be sought from them. Shared data can be assumed to be ones that are selected by the SQL query `SELECT c_1, c_2, ... FROM * WHERE ...` at the provider. Further, for each data item `c_1, c_2, ...,` it should be specified whether or not to return the result to the provider when the data is changed.
- In the BCDS, each site accepts requests for data provision from other sites. Define the data items that the site can provide to other sites in its data. For each data item shared with other sites, specify whether to return the result of data change at the provided sites (ReadWrite mode) or not to change data at the provided sites (ReadOnly mode). Further, a policy (update policy) for reflecting the change of the data item returned from the provided sites as an update of the data held by itself should be determined. Typically, the results of changes in the data returned from the provided sites are to be reflected in the data held on a first-come, first-served basis.

BCDS Agent configuration

The BCDS system consists of multiple BCDS Agents, each of which maintains its database and shares data through cooperation with the outside world. The BCDS system has a mechanism to automatically synchronize the data held by all the related BCDS Agents when one of the BCDS Agents updates data shared by multiple BCDS Agents to maintain consistency.

The BCDS Agent has an application function for its data, and also has a server function that provides (part of) its data in response to requests from other BCDS Agents, and a client function that requests the BCDS Agent for data-sharing. The server function and the client function are executed in parallel independently of the application. Here, the data held by itself includes, in addition to the data in the database owned by the BCDS Agent, shared data provided from other BCDS Agents by the client function. These data are represented in the form of tables in an SQL database, and the shared data items correspond to the columns in the table.

The data held by the BCDS Agent is updated not only by the application, but also by the client function when the changed data provided by other BCDS Agents, and the server when the data of ReadWrite mode returned from the provided BCDS Agents. Updates related to the client function and updates related to the server function are processed in parallel with the application by each function of the BCDS Agent, so there is no need to explicitly describe it in the application.

The BCDS Agent provides its data to other BCDS Agents as shared data by the server function. Some shared data items return the result when the provided shared data is changed by the destination (in ReadWrite mode), and others do not return the result even if changed by the destination (in ReadOnly mode).

- The data returned by changing the shared data of ReadWrite mode is reflected as an update of the data based on an updating policy defined by the BCDS Agent. If the same data item is provided to more than one BCDS Agent, not all changes will be reflected as updates to its data. In some cases, the change result at a certain destination may be discarded. This situation occurs because each of the BCDS Agents in the BCDS system allows data to be changed independently of each other. The handling of changes to shared data from multiple destinations and conflicts between them also depends on the updating policy.

The client function of BCDS Agent requests the data held by other BCDS Agents to be provided through the server function of each BCDS Agent. First, it establishes a connection for sending and receiving data by specifying data items to the providing BCDS Agent. At this time, it is specified whether each item is provided in the ReadWrite mode or the ReadOnly mode. If this differs from the service policy of the source BCDS Agent, the connection is rejected. Once the connection is granted, it will receive data thereafter through this connection.

- The client function of the BCDS Agent establishes a connection for the shared data requested by itself and receives data sent by the server function of the provider (when its data is updated). When the data item of the ReadWrite mode is changed by the application or the server function, the process of sending the data item to the provider is performed. Since these processes are performed in parallel with the application, there is no need to explicitly describe them in the application.

A BCDS system composed of the BCDS Agent server function and client function generally has a graph structure based on Peer-to-Peer connections. Data updates by BCDS Agent's applications are automatically propagated on this structure, and the consistency (Eventual Consistency) of the shared data on the system is maintained.

In the BCDS system, a new BCDS Agent may be added at any time, and the BCDS Agent in the system may be stopped at any time or some client functions may be stopped.