論文の内容の要旨

論文題目　　OS SUPPORTED DEPENDABLE SINGLE IP ADDRESS CLUSTER
(OS サポートによるディペンダブルな単一 IP アドレスクラスタ)

氏名　　藤田 肇

Internet servers have been becoming more important entity in today's society. Many services in the Internet depend on servers, and the services must not be disrupted at any moment. Therefore servers must be high performance, highly available, and highly reliable. In addition to that, servers should be efficient.

Single IP address cluster is a technique to implement a server by assigning a single IP address to a cluster of server nodes. In order to realize a server which meets some of the above requirements, single IP address cluster servers have been widely used.

In these cluster servers, some type of clusters, called broadcast-based clusters, are favorable in terms of its resiliency to node failures. In the broadcast-based clusters, a node failure does not affect the other server's communication. However, the existing broadcast-based clusters have limited load balancing capability. Thus they cannot fully utilize the total performance of server nodes under heavy loads.

This thesis first focuses on this load balancing issue and proposes a request dispatching method for realizing flexible load balancing on broadcast-based single IP address clusters. By applying this method, the performance and the efficiency of servers are improved. On the other hand, when considering a server node failure, there is an issue that some clients may not be able to connect to the server until the recovery process is completed. This issue becomes more crucial when the proposed request dispatching method is applied. Additionally, single IP address clusters in general do not provide any means to protect ongoing connection from server failures. In order to address these issues, this thesis also proposes two more methods, the Speculative SYN Packet Acceptance method and the Client-side Checkpointing method. The former method reduces the client-experienced latency when establishing a new connection under a node failure, and the latter method protects a connection from server crash. These methods improve availability and reliability of a server.

As described above, this thesis proposes three methods to improve server dependability. The first method is a request dispatching method for load balancing in broadcast-based clusters. Both layer-4 and layer-7 request dispatching mechanisms are designed and implemented. By making connection dispatching decisions at specific scheduler nodes, it becomes possible to employ flexible load balancing algorithms, which is difficult for the existing broadcast-based systems. Because the scheduler node is involved in the communication only when a connection is being established, its failure does not affect other nodes after the connection is established. The least connection scheduler for the layer-4 load dispatching, and the locality-aware scheduler for the layer-7 dispatching, are implemented and evaluated using the SPECweb2005 benchmark. The evaluation shows that the proposed method processed 13% more requests than the existing method in the layer-4 load balancing, and reduced the number of slow downloads, which failed the speed criteria, by 48% in the layer-7 load balancing.

The second proposed method is the Speculative SYN Packet Acceptance mechanism, in which the secondary node speculatively accepts the SYN packet from the client and responds in order to reduce the TCP connection latency seen from the client
in case of node failure. Experiments show that this method ensures that a server cluster responds to clients almost within 0.4 second even there is a failed node, whereas it takes 3 to 9 seconds without the proposed method.

The third method is the client side checkpointing method. A server application extracts its essential per-connection state information and stores it to the client's memory. When a server crashes, the client automatically reconnects to the server cluster and the server reconstructs the state from the checkpoint stored in the client. Benchmark tests using the ffserver, a media streaming server, shows that the performance overhead of the proposed method is as low as less than 4%, and the memory overhead is less than twice of that of the normal version.

インターネットサーバは今日の社会にとってますます重要な存在となっている。インターネット上の多くのサービスはサーバに依存しており、それらのサービスはいかなる場合も中断してはならない。よってサーバは高性能で、可用性が高く、また信頼性が高いことが求められる。それらに加えて、サーバは高効率でなければならない。

これらの要求の達成に近づくため、複数のサーバノードからなるクラスタに 1 つの IP アドレスを割り当ててサーバとして用いる、単一 IP アドレスクラスタという手法が広く使われてきた。その中でも、ブロードキャスト型とよばれる単一 IP アドレスクラスタは、あるサーバノードの故障が他のノードの通信に影響しないという特徴があり、耐故障性の観点から望ましい方式である。しかし、既存のブロードキャスト型クラスタではクライアントからのリクエストを各サーバノードに分配するための負荷分散手法が制限されており、高負荷時に全サーバノードの能力を活用しきれない。

そこで本論文ではまずこの負荷分散に焦点をあて、ブロードキャスト型クラスタにおいて柔軟な負荷分散を行うためのリクエスト分散機構を提案する。これによって、サーバの高性能化、同時に高効率化を実現する。一方で、サーバノードの故障を考えると、ブロードキャスト型クラスタにおいても故障から復旧措置が行われるまでの間はサーバに接続できないクライアントが発生するという問題がある。この問題は、本論文で提案するリクエスト分散機構を導入するとより顕著となる。さらに、単一 IP アドレスクラスタ全般は故障発生時にそのサーバノードで実行中だった処理の保護については解決手段を提供しておらず、実行途中の処理は失われてしまう。これらの問題に対処するため、本論文ではさらに、サーバノード故障時にも接続確立に要する遅延を削減するための投機的 SYN パケット受信機構と、サーバアプリケーションの状態をクライアント側に保存しサーバ故障に備えるクライアントサイドチェックポインティング機構を提案する。これら 2 つの手法によって、サーバの可用性と信頼性を向上させる。

本論文では、上述の通り、サーバのディペンダビリティ向上のための 3 つの手法を提案する。1 つめの手法は、ブロードキャスト型クラスタにおいて負荷分散を行うための、レイヤ 4 およびレイヤ 7 リクエスト分散機構である。これはコネクション割り当ての判断を特定のスケジューラノードで行うことにより、旧来のブロードキャスト型クラスタでは難しかった、柔軟な負荷分散アルゴリズムを採用可能にするものである。スケジューラノードは TCP 接続の開始時にのみ関与するため、接続の確立後にスケジューラノードが故障したとしても確立済通信には影響を及ぼさない。レイヤ 4 負荷分散のために最小コネクション数スケジューラ、レイヤ 7 負荷分散のために局所性を考慮したスケジューラを実装した。SPECweb2005 を用いた評価において、レイヤ 4 負荷分散では既存手法に比べ 13%多くのリクエストを処理し、レイヤ 7 負荷分散では基準を満たさない遅いダウンロード数を 48%削減した。

2 つめの提案手法は投機的 SYN パケット受信機構である。これは、セカンダリとなっているノードが投機的にクライアントからの SYN パケットを受信して返答することによって、サーバノード故障時に TCP 接続確立に要する遅延を削減するものである。サーバノードの

故障を模擬する実験の結果、提案手法を用いないとリクエスト完了までに 3〜9 秒程度の遅延を観測するクライアントが多数存在したのに対し、提案手法を用いると遅延を概ね 0.4 秒以内に収めることができることが示された。

3 つめの提案手法は、クライアント側にチェックポイントを取る手法である。サーバアプリケーションが自らコネクションごとに必要な状態情報を切り出し、クライアントのメモリに保存する。サーバがクラッシュした際は、クライアントは自動的にサーバクラスタに対して再接続し、サーバはクライアントに保存してあったチェックポイントを用いて状態を復元する。メディアストリーミングサーバ ffserver を用いたベンチマークでは、提案手法の性能オーバヘッドは 4%未満と低く、メモリ使用量ももともとのアプリケーションの 2 倍を超えないことが示された。