

Arcus Cloud 기본 사항

Arcus는 확장된 key-value 데이터 모델을 제공한다. 하나의 key는 하나의 데이터만을 가지는 simple key-value 유형 외에도 하나의 key가 여러 데이터를 구조화된 형태로 저장하는 collection 유형을 제공한다.

Arcus cache server의 key-value 모델은 아래의 기본 제약 사항을 가진다.

- 기존 key-value 모델의 제약 사항
 - Key의 최대 크기는 250 character이다.
 - Value의 최대 크기는 1MB이다.
- Collection 제약 사항
 - 하나의 collection이 가지는 최대 element 개수는 50,000개이다.
 - Collection element가 저장하는 value의 최대 크기는 4KB이다.

아래에서 Arcus cloud를 이해하는 데 있어 기본 사항들을 기술한다.

- [서비스코드](#)
- [Arcus Admin](#)
- [Cache Key](#)
- [Cache Item](#)
- [Expiration, Eviction, and Sticky Item](#)

서비스코드

서비스코드(service code)는 Arcus에서 cache cloud를 구분하는 코드이다. Arcus cache cloud 서비스를 응용들에게 제공한다는 의미에서 "서비스코드"라는 용어를 사용하게 되었다.

하나의 응용에서 하나 이상의 Arcus cache cloud를 구축하여 사용할 수 있다. Arcus java client 객체는 하나의 Arcus 서비스코드만을 가지며, 하나의 Arcus cache cloud에만 접근할 수 있다. 해당 응용이 둘 이상의 Arcus cache cloud를 접근해야 한다면, 각 Arcus cache cloud의 서비스코드를 가지는 Arcus java client 객체를 따로 생성하여 사용하여야 한다.

Arcus Admin

Arcus admin은 ZooKeeper를 이용하여 각 서비스 코드에 해당하는 Arcus cache cloud를 관리한다. 특정 서비스 코드에 대한 cache server list를 관리하며, cache server 추가 및 삭

제에 대해 cache server list를 최신 상태로 유지하며, 서비스 코드에 대한 cache server list 정보를 arcus client에게 전달한다. Arcus admin은 highly available하여야 하므로, 여러 ZooKeeper 서버들을 하나의 ZeeKeeper ensemble로 구성하여 사용한다.

Cache Key

Cache key는 Arcus cache에 저장하는 cache item을 유일하게 식별한다. Cache key 형식은 아래와 같다.

Cache Key : [<prefix>:]<subkey>

- <prefix> - Cache key의 앞에 붙는 namespace이다.
 - Prefix 단위로 cache server에 저장된 key들을 그룹화하여 flush하거나 통계 정보를 볼 수 있다.
 - Prefix를 생략할 수 있지만, 가급적 사용하길 권한다.
- delimiter - Prefix와 subkey를 구분하는 문자로 default delimiter는 콜론(':')이다.
- <subkey> - 일반적으로 응용에서 사용하는 Key이다.

Prefix와 subkey는 아래의 명명 규칙을 가진다.

- Prefix는 영문 대소문자, 숫자, 언더바(_), 하이픈(-), 플러스(+), 점(.) 문자만으로 구성될 수 있으며, 이 중에 하이픈(-)은 prefix 명의 첫번째 문자로 올 수 없다.
- Subkey는 공백을 포함할 수 없으며, 기본적으로 alphanumeric만을 사용하길 권장한다.

Cache Item

Arcus cache는 simple key-value item 외에 다양한 collection item 유형을 가진다.

- simple key-value item - 기존 key-value item
- collection item
 - list item - 데이터들의 linked list를 가지는 item
 - set item - 유일한 데이터들의 집합을 가지는 item
 - b+tree item - b+tree key 기반으로 정렬된 데이터 집합을 가지는 item

Expiration, Eviction, and Sticky Item

각 cache item은 expiration time 속성을 가진다. 이 값의 설정으로 자동 expiration을 지정하거나 expire되지 않도록 지정할 수 있다.

Arcus cache는 memory cache이며, 한정된 메모리 공간을 사용하여 데이터를 caching한다. 메모리 공간이 모두 사용된 상태에서 새로운 cache item 저장 요청이 들어오면, Arcus cache는 "out of memory" 오류를 내거나 LRU(least recently used) 기반으로 오랫동안 접근되지 않은 cache item을 evict시켜 available 메모리 공간을 확보한 후에 새로운 cache item을 저장한다.

특정 응용은 어떤 cache item이 expire & evict 대상이 되지 않기를 원하는 경우도 있다. 이러한 cache item을 sticky item이라 하며, expiration time을 -1로 설정하면 된다. Sticky item의 삭제는 전적으로 응용에 의해 관리되어야 함을 주의해야 한다. 그리고, sticky item 역시 메모리에 저장되기 때문에 server가 restart되면 사라지게 된다.