

CCNSimの使い方

東京工科大学コンピュータサイエンス学部 専任講師

金光 永煥(Hidehiro KANEMITSU)

kanemitsuh@stf.teu.ac.jp

はじめに

- アルゴリズムの作成・反映手順については <https://docs.google.com/document/d/15F-Jyic4gU-P5o8CX7rdpCiahWO5fXJPBLTWctt5FMg/edit?pli=1> にあります.
- このスライドでは, クラスとAPIについて説明します.
- 主に,

- CCNRouter: ルータであり, パケット転送, キャッシング
- CCNNode: オリジナルコンテンツ保持, コンテンツ要求
- CCNMgr: CCNRouter / CCNNodeを生成・管理

があります.

大事なクラス：CCNMgr

- ノード、ルータを管理するクラスであり、シングルトン。
- このクラスは、インスタンス化せずに直接、アクセスできる。（CCNMgr.getIns().**メソッド名**でアクセス）
- ノードやルータは、↓のハッシュマップ（ID, オブジェクト）で保持。
 - ノード集合: `HashMap<Long, CCNNode> nodeMap`
 - つまり、外部からはCCNMgr.getIns().getNodeMap()で、この集合を取得できる。
 - ルータ集合: `HashMap<Long, CCNRouter> routerMap`
 - つまり、外部からはCCNMgr.getIns().getRouterMap()で、この集合を取得できる。

大事なクラス：CCNNode

- **CCNNodeクラス(AbstractNodeを継承)** .
 - IDには, NodeID(Long型) をもつ.
 - 自身が最初から持つコンテンツ領域 :
 - HashMap<String, CCNContents> **ownContentsMap**
 - 取得したコンテンツ領域 :
 - LinkedBlockingQueue<CCNConents> **contentsQueue**.
 - これは, AbstractNodeで定義されている.
 - あとは, type: (ルータ/ノード), usedRouting(用いるFIBルーティングアルゴリズム), receiver(データ受信処理用スレッド)をフィールド変数に持つ.

大事なクラス: CCNRouter

- CCNRouterクラス (AbstractNodeを継承)

- IDはLong routerID, FIBは, FIB FIBEntry, PITは, PIT PITEntry, CSは, CS CSEntryとして保持.
- Faceリストは, ルータ宛のもの(face_routerMap)とノード宛(face_nodeMap)の二種類ある.
 - いずれもHashMap<Long, Face>であり, <FaceID, Faceオブジェクト>.
 - 肝心の宛先IDは, Face.pointerIDにセットされる (ルータIDやノードID)
 - Face.type(ノードだと0, ルータだと1) で, あて先のタイプを識別している.

FIBについて

- < Prefix, Faceリスト > の情報を持つ. 実際には,
 - `HashMap<String, LinkedList<Face>> table;`
を保持している.
- Faceには,
 - Long FaceID,
 - Long pointerID (宛先ルータ / ノードのID)
が入る.

CSについて

- キャッシュを保持するためのクラスであり,
 - `HashMap<String, CCNContents> cacheMap`
のハッシュマップを保持している。形式は, <prefix, コンテンツ>である。