

DIFS 2020

plan

{uni2u}{labry}@etri.re.kr
Network Research Division
Data Centric Networking lab



Abstract

- OpenSource 추진 (Named Data Networking)
 - 기본 기능 정리
 - 부족한 기능 보완
 - flow 체크 (정상 동작 확인; NDN flow 체크)
 - 인터페이스 추가
- DIFS paper (성능 및 기능)
 - sqlite3 VS filesystem
 - signature
 - layered hash tree

기본 기능

DIFS basic command and flow



기본 기능

- 현재 제공되는 get/put 이외 필수 기능 추가
- repo-ng 의 redmine issue (<https://redmine.named-data.net/projects/repo-ng/issues/report>) 기반 기본 기능 체크
 - issue 내용을 파악하고 필요한 기능은 DIFS 에 적용
 - 예) delete 기능 등
- repo-ng 동작 flow 체크
 - 예) Automatic per-data-packet (per-group) prefix registrations (<https://redmine.named-data.net/issues/4247>)
 - 저장한 모든 data 의 prefix 를 local NFD 에 advertise 하는 것이 완료된 것으로 확인됨
 - 모든 prefix 를 advertise 하는 방식인지 code 에서 체크
 - data name = network name + data name 형식인지?
 - network name 은 같은 도메인에서 필요하지 않음
 - 즉, 저장한 data name 이 어디까지가 prefix 로 지정되는지, 해당 prefix 는 local NFD 에 advertise 되는지 체크



기본 기능: repo-ng redmine issue

- issue 내용 중 반드시 구현 하여야 할 내용 정리
 - 리스트 내용을 확인 후 DIFS 에 필요한 기능 추출
 - 회의를 통하여 기능 리스트 업
 - 기본적으로 get/put/delete 기능 구현
 - 그외 추가 기능 구현
- 기본 기능 체크
 - closed 의 경우 repo-ng flow 확인
 - repo-ng flow 확인이 완료되면 DIFS 확인
 - new/in progress 의 경우 DIFS 구현
 - 구현 후 DIFS flow 확인

유형


	진행중	완료됨	합계
Task	15	24	39
Bug	7	15	22
Feature	5	5	10

우선순위

	진행중	완료됨	합계
Immediate	-	-	-
Urgent	-	3	3
High	5	6	11
Normal	21	34	55
Low	1	1	2

#	유형	상태	우선순위	제목	담당자	변경
4634	Feature	In Progress	Normal	Forwarding hint support	weijie yuan	2019/05/15 15:14 ...
4247	Feature	Closed	High	Automatic per-data-packet (per-group) prefix registrations	Nicholas Gordon	2017/10/23 08:53 ...
4129	Feature	New	Normal	Management Dispatcher for repo commands	Muktadir Chowdhury	2018/10/27 12:50 ...
3768	Feature	New	Normal	Storage management / quota features		2016/09/07 07:55 ...
3767	Feature	New	High	Provide option to enable prefix caching behavior		2016/09/07 07:53 ...
3766	Feature	New	High	Enable external command to trigger prefix registration with local NFD		2017/06/08 10:04 ...
1791	Feature	Closed	Normal	ndnwatchfile: Tool to execute repo watch protocol	WeiQi Shi	2017/06/08 09:54 ...
1784	Feature	Closed	Normal	New Insert Protocol: Watching prefix		2017/06/08 09:53 ...
1778	Feature	Closed	Normal	Configuration to enable and disable validation		2017/06/08 09:49 ...
1485	Feature	Closed	Normal	Backdoor to insert data packets into repo	Alex Afanasyev	2014/04/19 21:31 ...

Feature #4247

**Automatic per-data-packet (per-group) prefix registrations**« 뒤로 | 2/10 | 다음 »

Alex Afanasyev이(가) 2년 이상 전에 추가함. 2년 이상 전에 수정됨.

상태:	Closed	시작시간:	2017/03/13
우선순위:	High	완료기한:	
담당자:	Nicholas Gordon	진척도:	<div><div></div></div> 80%
목표버전:	sqlite3	추정시간:	

설명

After the switch to v2::Certificate, we need to "restore" ability to serve certificates from the repo on a passing by node that potentially has those data packets. In order to do so, repo needs to automatically register with local NFD all (some) prefixes of data.

This is a similar function to what we previously (a long time ago) discussed for PIB service.

Signature

One Signature



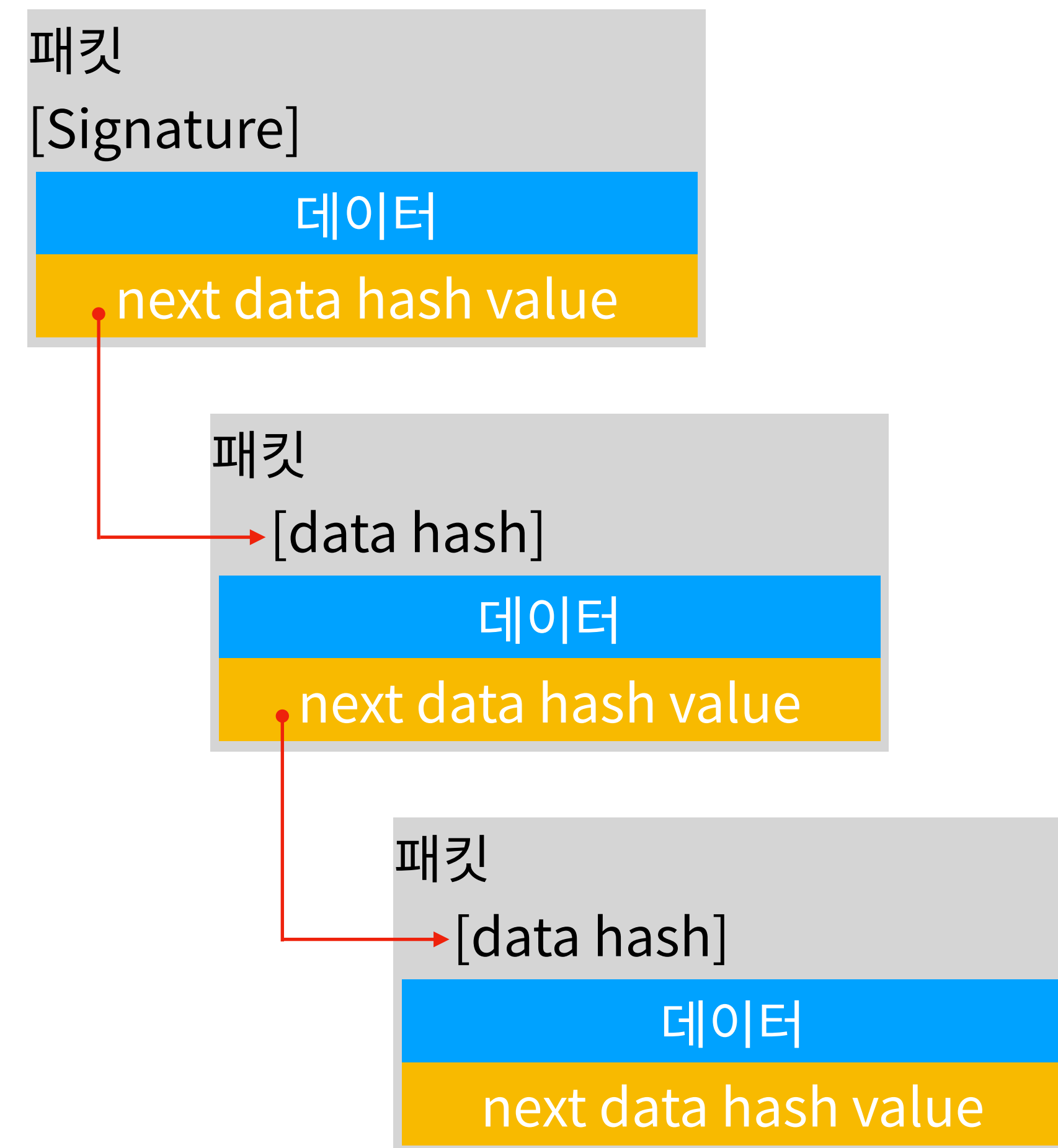
Signature

- NDN 은 모든 패킷에 대한 ‘Signature’ 요구 (NDN 에서 인정하는 문제점)
- Signature 를 줄이는 방안
 - embedded manifest 등의 방안이 소개됨
- DIFS 도 데이터가 저장되는 저장소로서 ‘Signature’ 에 대한 동일한 이슈
 - 2019년 DIFS 는 ‘DigestSha256’ 을 채택함
 - 2020년 ‘blake’ 기반의 hash-chain 구조



Signature: Hash Chain (integrity verifying, originality)

- DIFS 에서 제공할 integrity verifying 방안
 - One Signature 기반 모든 데이터 무결성 체크 방법
- 동작 절차
 - 최초 데이터 패킷은 signature 와 함께 digest 로 바로 뒤 데이터의 hash 포함
 - 두번째 데이터 패킷은 digest 로 바로 뒤 데이터의 hash 포함
 - 두번째 데이터 패킷을 받으면 첫번째 데이터를 받으면서 포함된 digest (두번째 데이터의 hash) 와 실제 받은 데이터의 hash 비교
 - ...
 - 체인 형식으로 제공



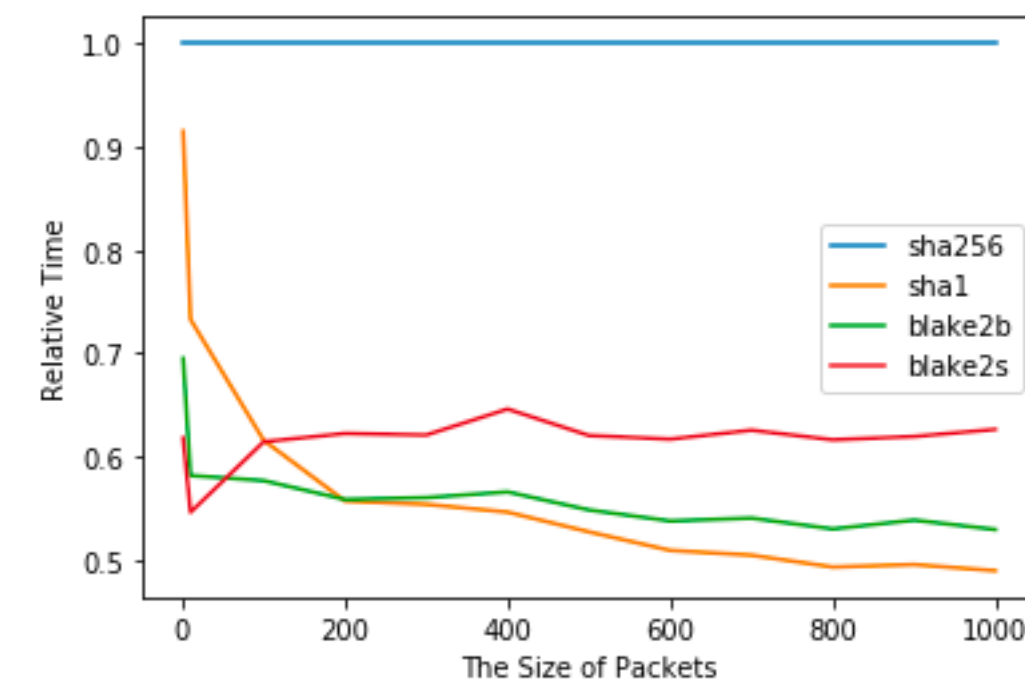


Signature: using 'blake' signature type

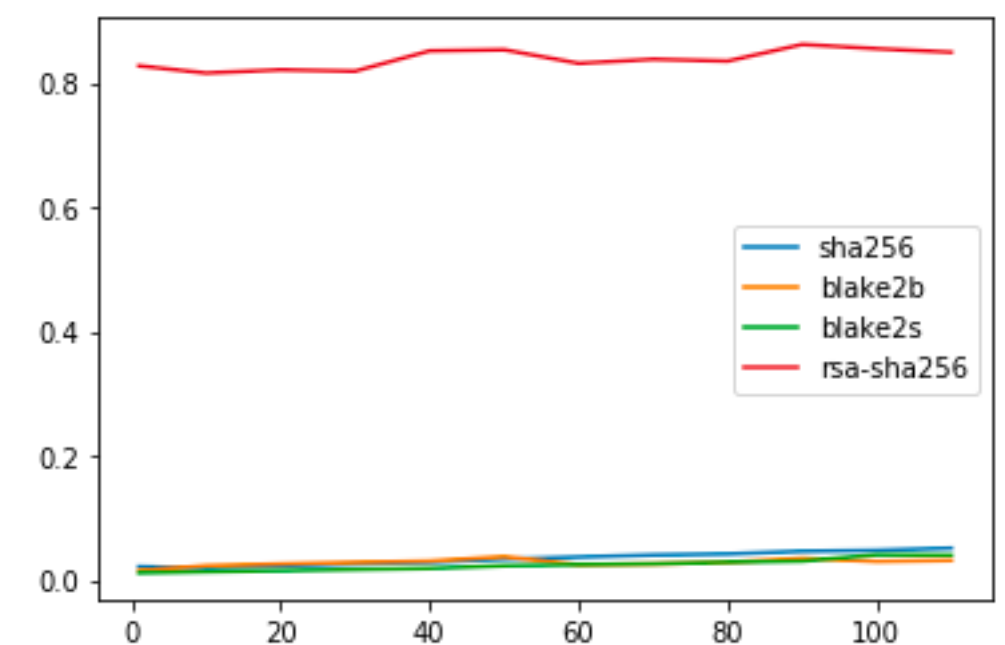
- Signature 성능을 높이기 위함
 - sha256 보다 성능을 대폭 개선한 'blake' 활용
- 'sha256' 자체 성능 문제 해소
 - blake2b 압도적으로 우수함
- 적용 방안
 - NDN Packet Format Signature Type 추가
 - <https://named-data.net/doc/NDN-packet-spec/current/signature.html>
 - 'SignatureBlakeWithRsa' 또는 'DigestBlake' 등

Value	Reference	Description
0	DigestSha256	Integrity protection using SHA-256 digest
1	SignatureSha256WithRsa	Integrity and provenance protection using RSA signature over a SHA-256 digest
3	SignatureSha256WithEcdsa	Integrity and provenance protection using an ECDSA signature over a SHA-256 digest
4	SignatureHmacWithSha256	Integrity and provenance protection using SHA256 hash-based message authentication codes
2,5-200		reserved for future assignments
>200		unassigned

→ 'blake' type 추가



[sha256, sha1, blake2b, blake2s 성능 비교]



[rsa, sha256, blake2b, blake2s 성능 비교]

현재 개념 정리가 완료됨
세부 내용은 정리중

Layered Hash Tree

Fast provenance and integrity check

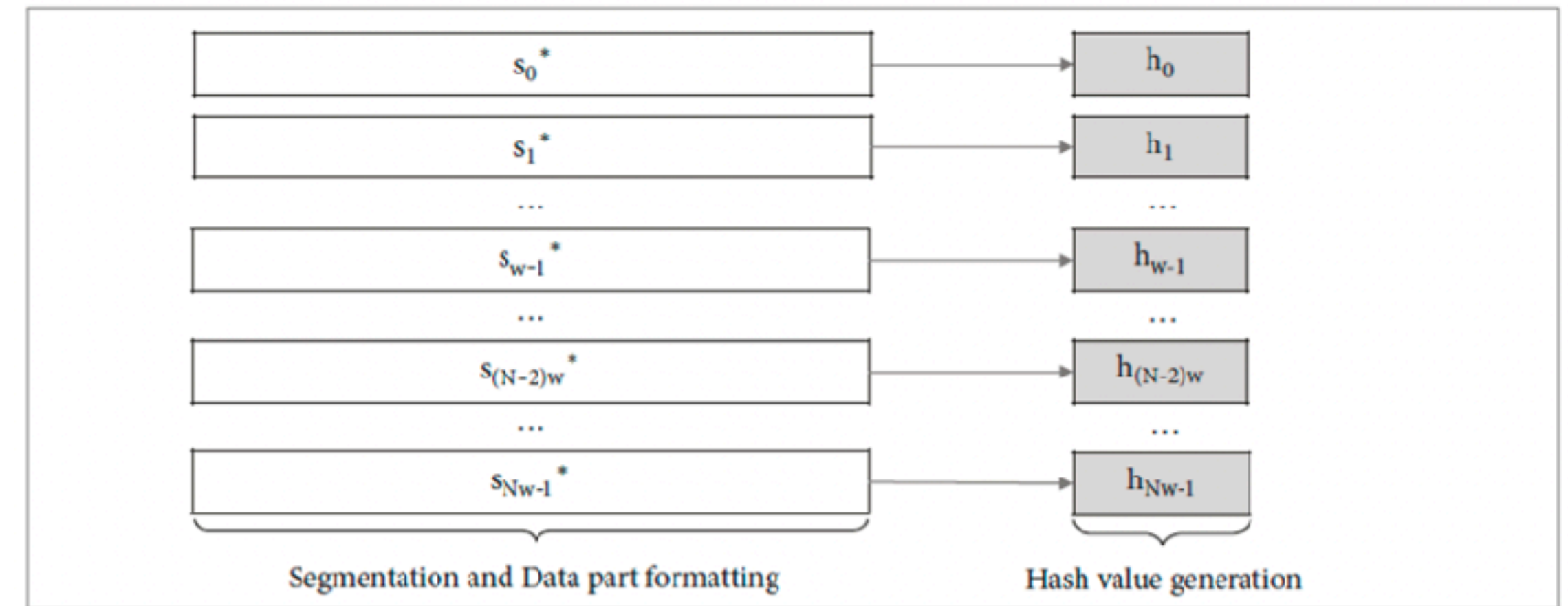


Layered Hash Tree

- 데이터 저장소에 파일 저장
 - NDN 입장에서 신규 데이터가 생성
 - NDN 데이터의 버전 관리 필요
 - 데이터 전체에 대한 calculation overhead 를 줄일 필요
 - Hash-Chain 을 사용하는 경우
 - 새로운 데이터가 뒤에 added 되는 경우 (version up)
 - chain 구성을 위해 새로 hash chain 을 만들어야 함
- TLDA (Two Layered Data Authentication) 메커니즘 활용

- 기본적으로 2 단계 진행

① Data part encoding(segmentation, formatting, and hash value generation)



② Meta part encoding(formatting and signing)

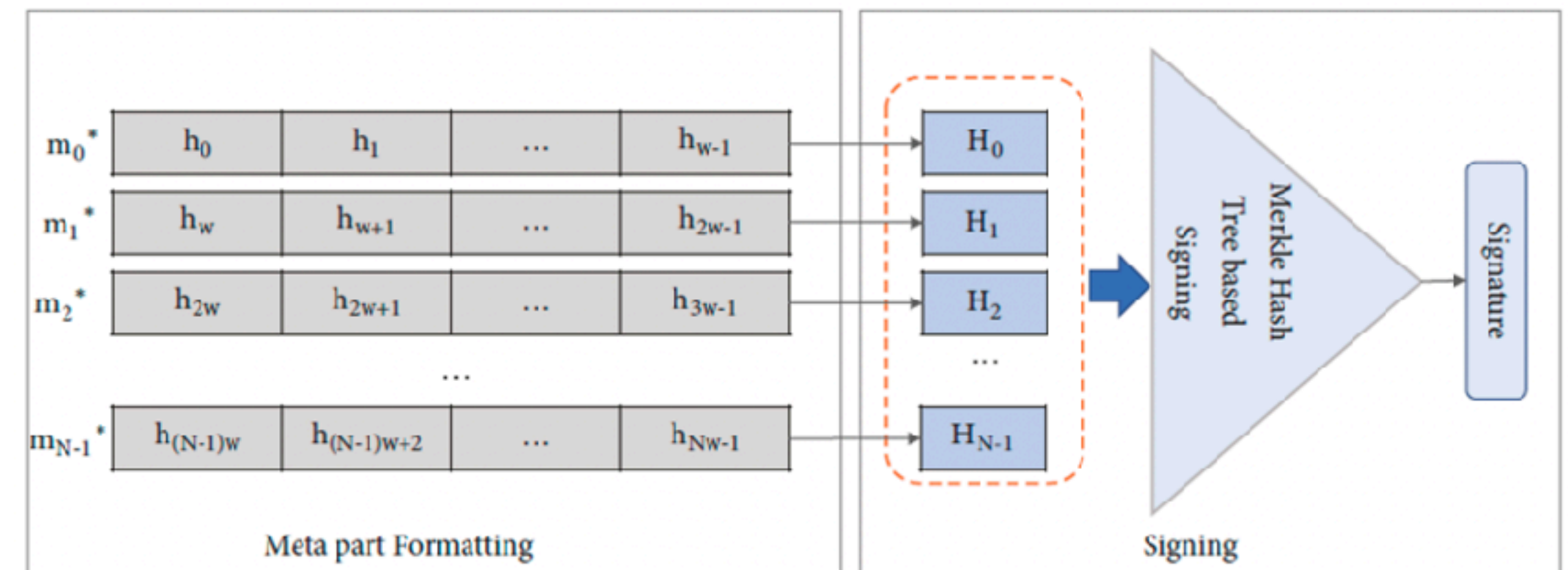


FIGURE 1: Process for generating authentication information in TLDA.

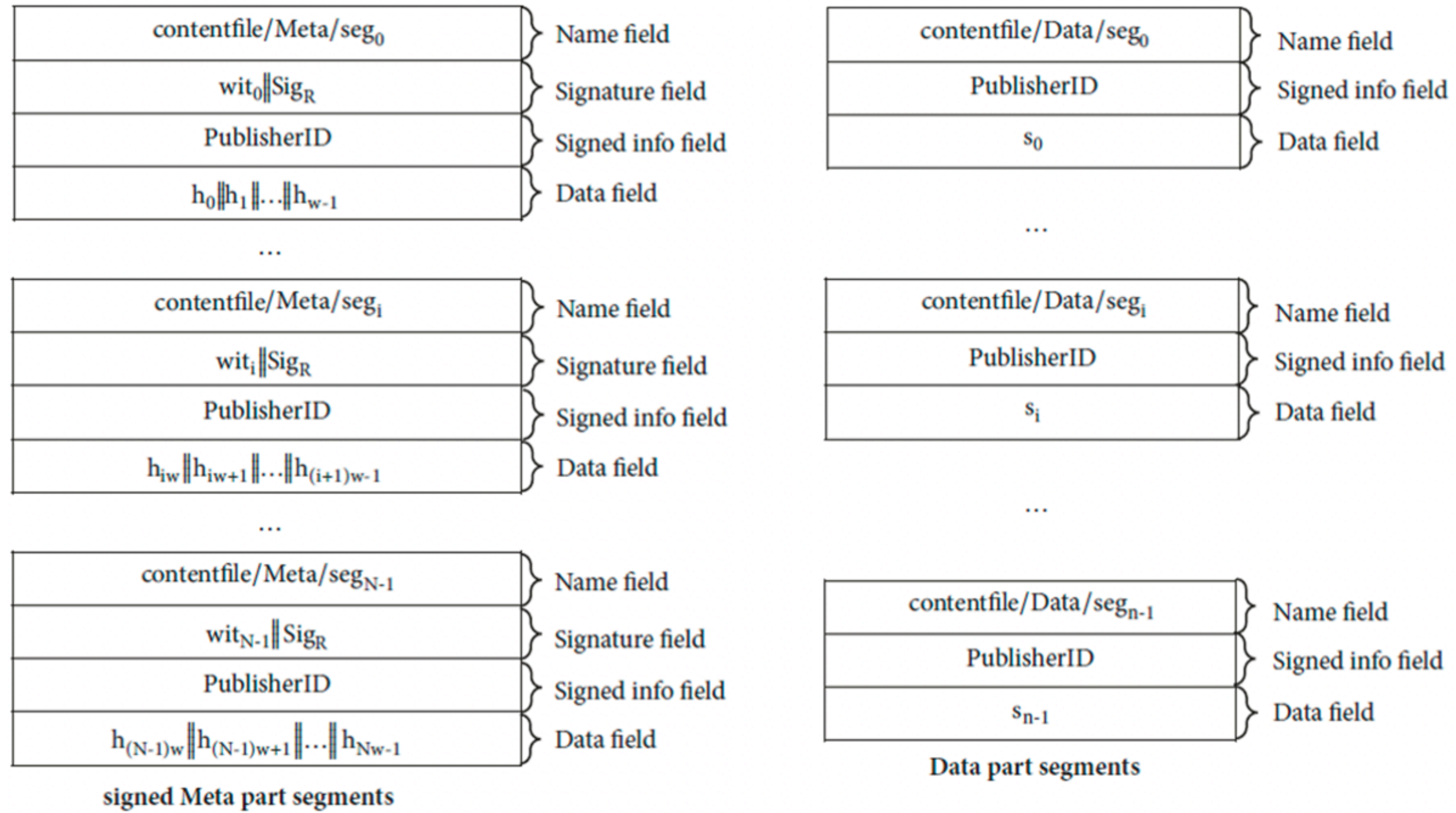


FIGURE 5: Example of the proposed format for segments in encoded Data part and encoded Meta part.

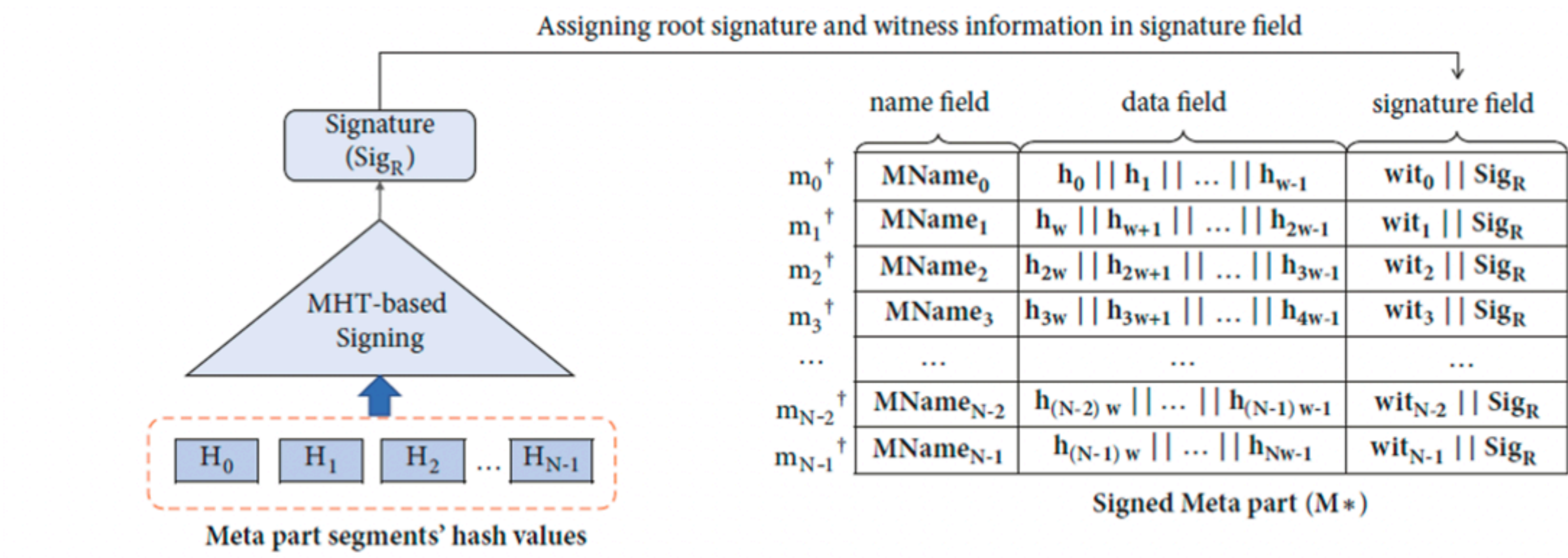


FIGURE 4: Meta part encoding: MHT-based signing. (For simple description, we omit signed info field in each segment in this figure).

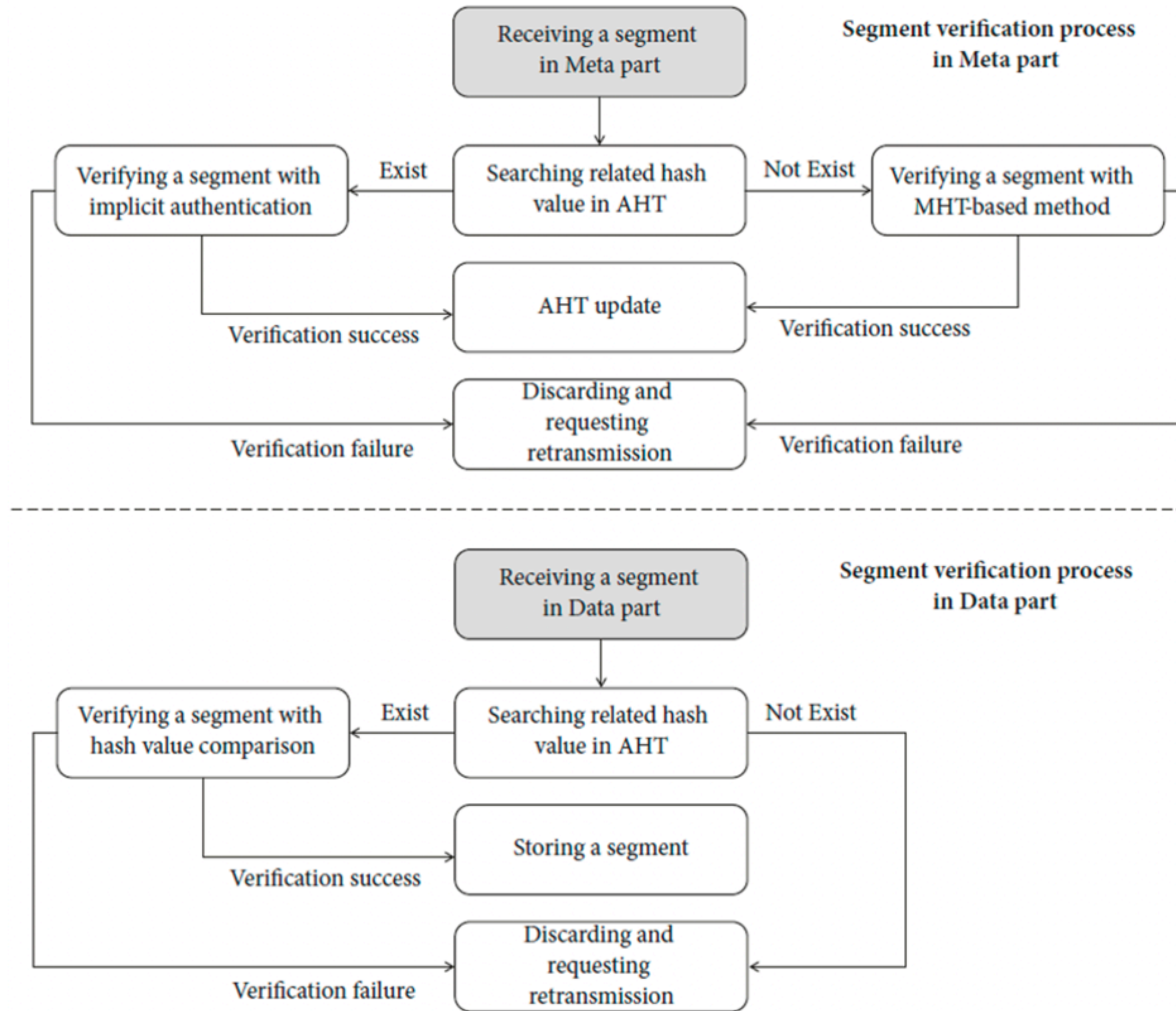


FIGURE 6: Proposed content segment verification process.

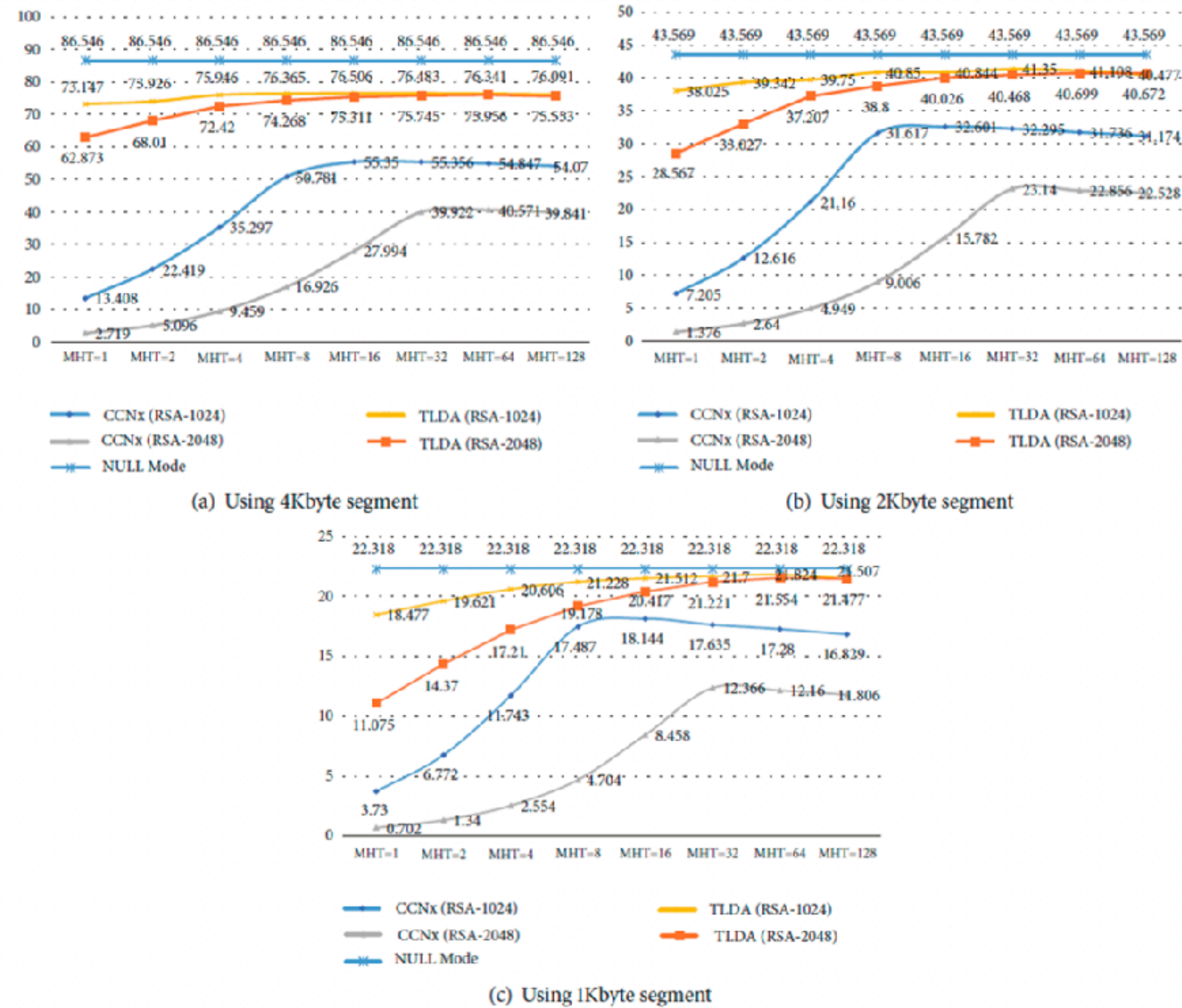


FIGURE 7: Transmission performance comparison when transmitting a 128Mbyte content with segment size of 4Kbytes, 2Kbytes, and 1Kbytes. (x-axis is MHT size applied to both CCNx and TLDA and y-axis is Mbps.)

TABLE 3: Time comparison for RSA signing, verifying, and hashing to a segment of 4Kbytes, 2Kbytes, and 1Kbytes. (Timings are the average of one million executions.) RSA signature operation makes use of SHA-256 for message hashing.

	Segment Size		
	4Kbytes	2Kbytes	1Kbytes
RSA 1024 Signing	1.9236 <i>ms</i>	1.8659 <i>ms</i>	1.7832 <i>ms</i>
RSA 1024 Verifying	0.3508 <i>ms</i>	0.2955 <i>ms</i>	0.2624 <i>ms</i>
RSA 2048 Signing	11.2315 <i>ms</i>	10.9907 <i>ms</i>	10.8255 <i>ms</i>
RSA 2048 Verifying	0.7487 <i>ms</i>	0.6983 <i>ms</i>	0.6753 <i>ms</i>
SHA-256	0.05071 <i>ms</i>	0.02556 <i>ms</i>	0.01329 <i>ms</i>

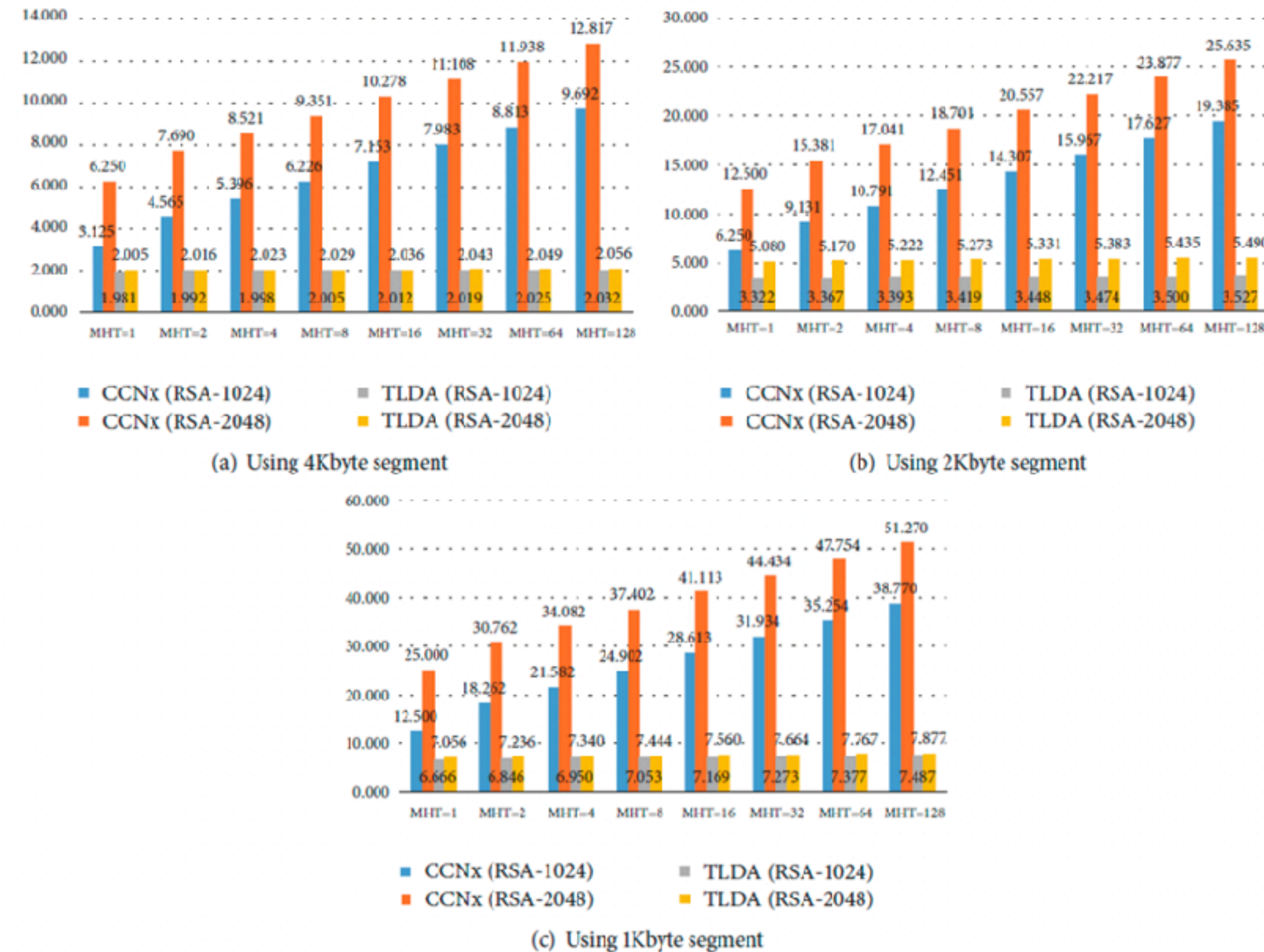
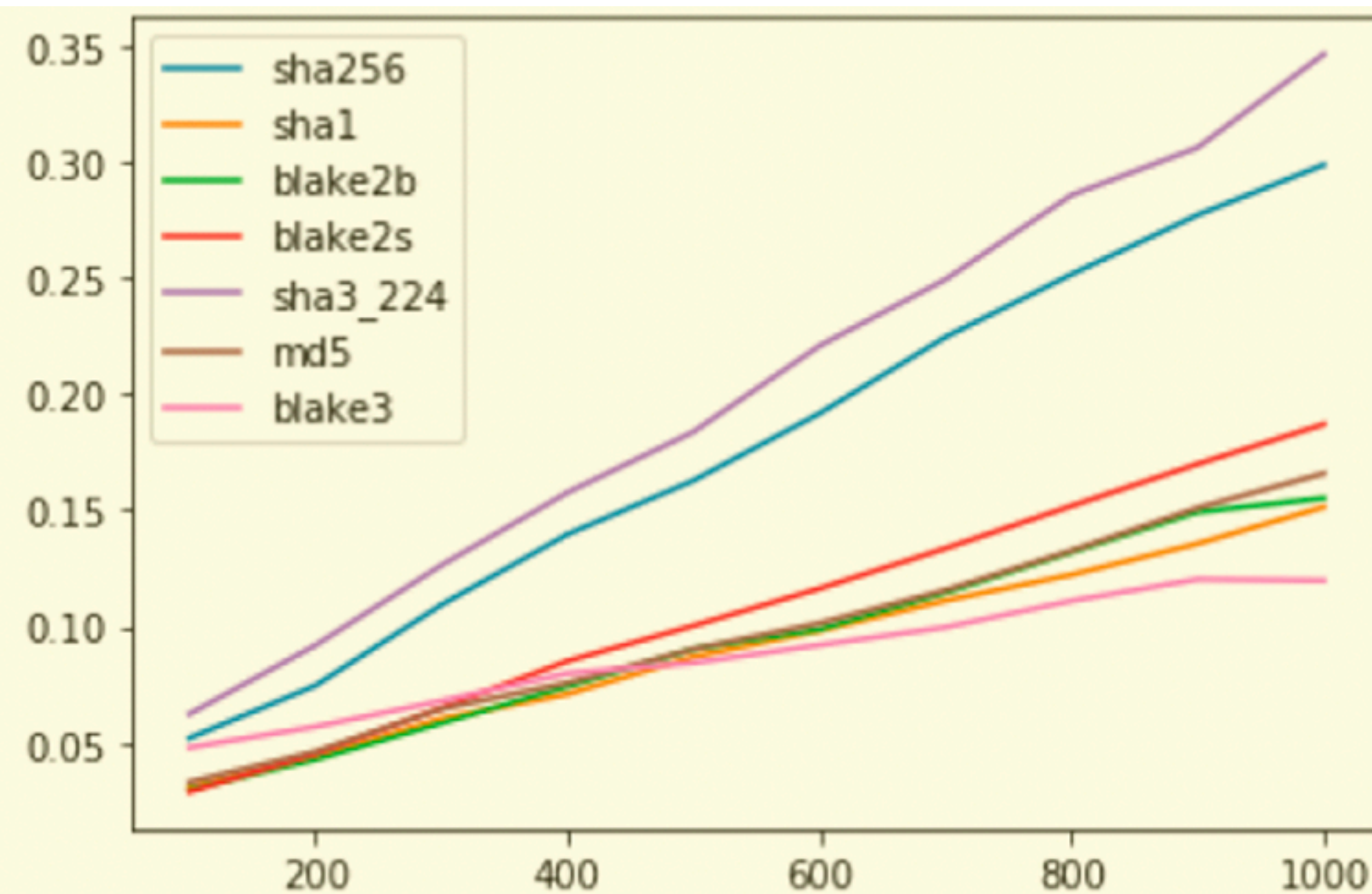


FIGURE 9: Communication overhead comparison when transmitting a 128Mbyte content with segment size of 4Kbytes, 2Kbytes, and 1Kbytes. (x-axis is MHT size applied to both CCNx and TLDA and y-axis is the introduced communication overhead.)



Sha-256 is used as the baseline performance measurement unit and the Y-axis shows the relative time they needed to perform a hashing. X-axis show the packet size and the unit is 10 bytes, meaning 200 denotes 2k bytes, 400 4k bytes and so on.

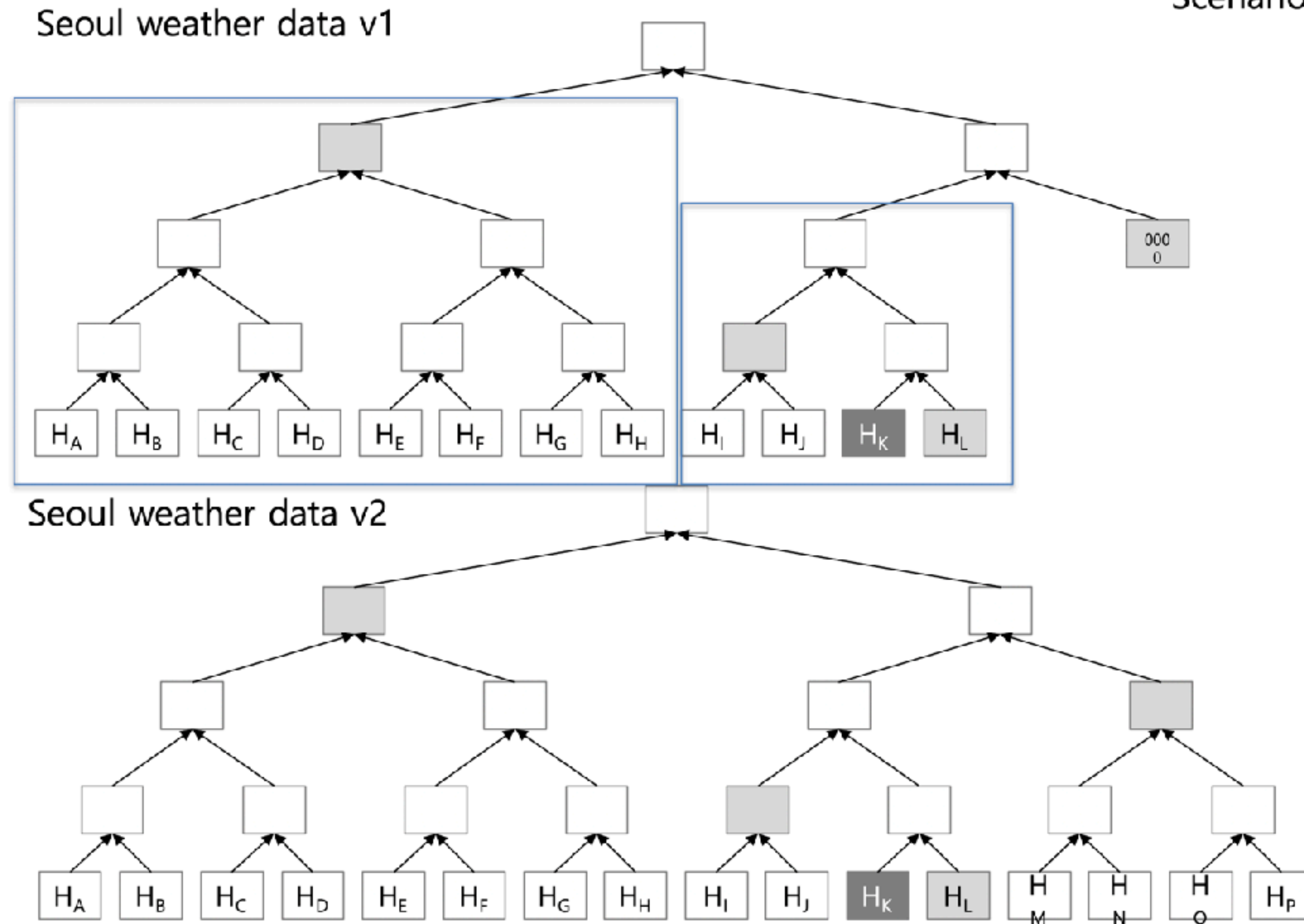
- 15MB, 128MB 정도의 작은 파일 비교만 적용
 - DIFS 의 경우 700MB, 1GB, 10GB 등의 큰 파일을 대상으로 함
- source code 제공되지 않음
- 인증 관련 기능의 부재
 - 여러 조각으로 나뉘어 제공되는 경우 조각마다 signature 적용 부담



시나리오 #1

The newer version has the same depth and adds new segments to the paddings.

Scenario #1

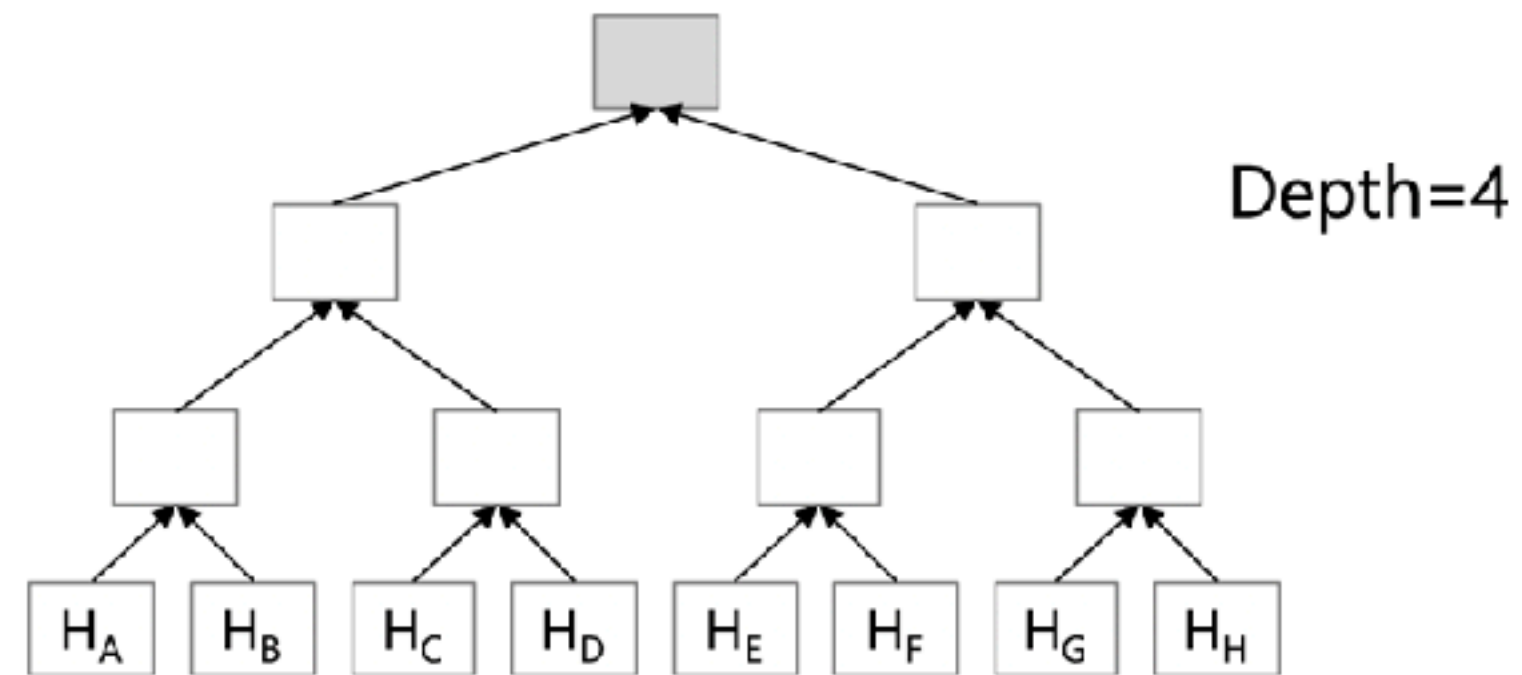




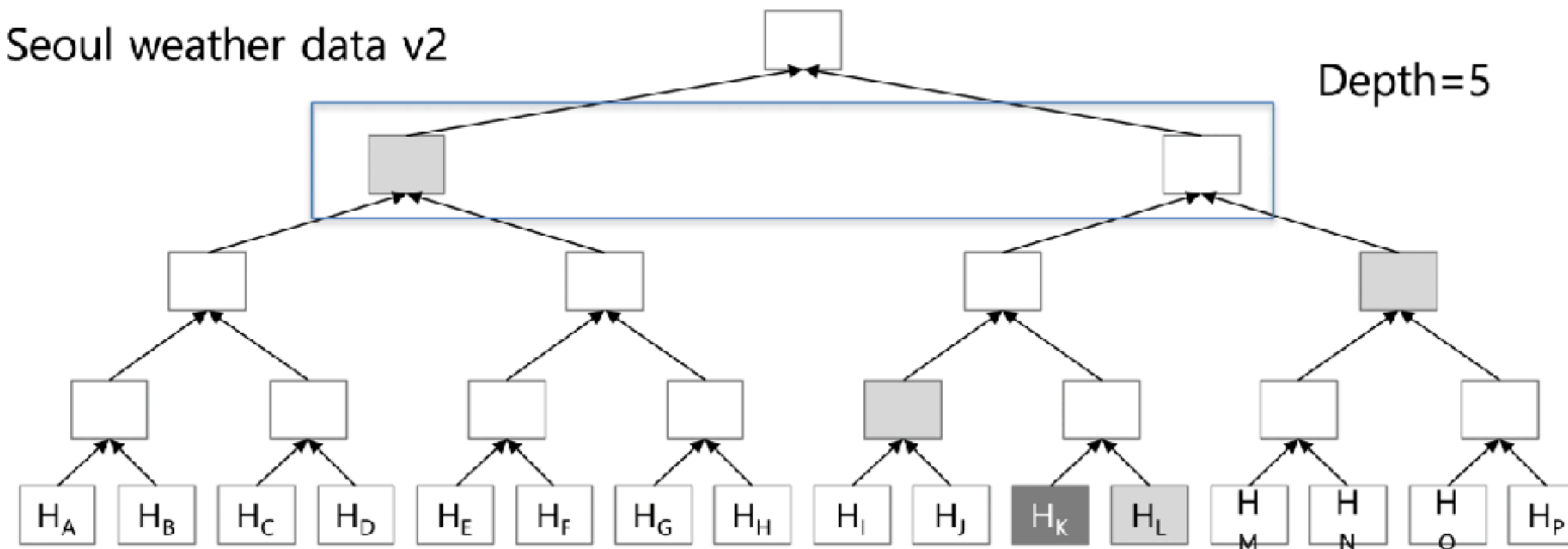
The newer version includes the older version as an subtree.

Scenario #2

Seoul weather data v1



Seoul weather data v2

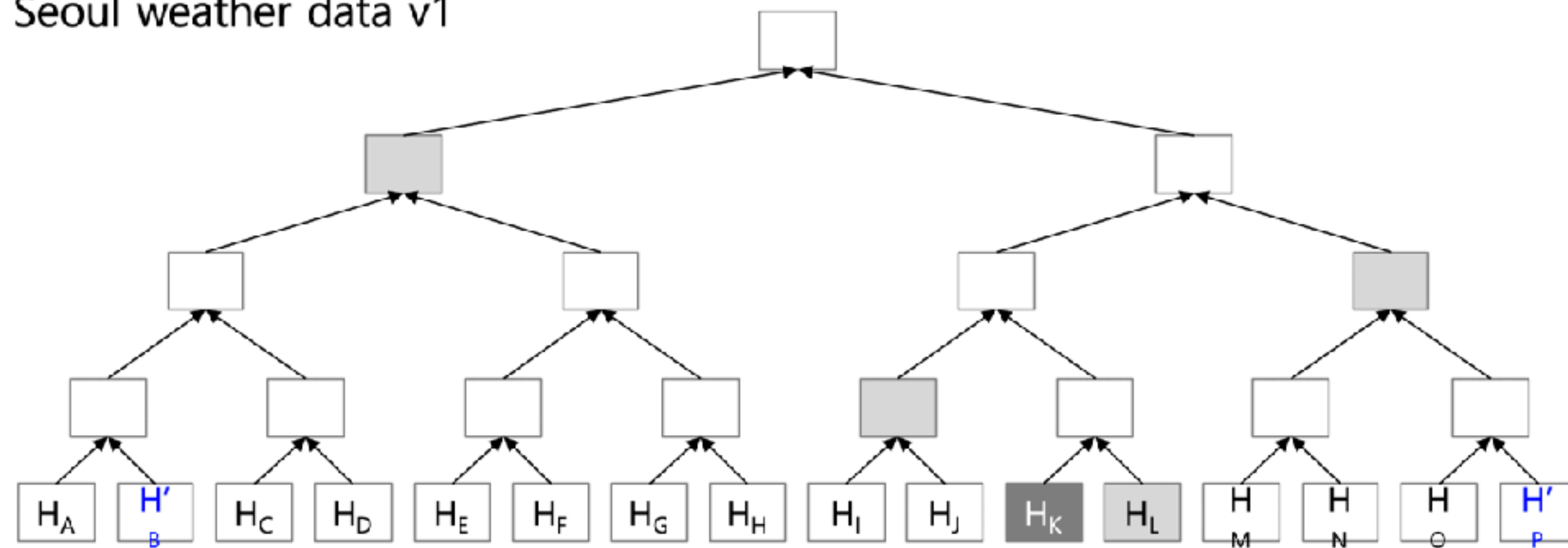




시나리오 #3

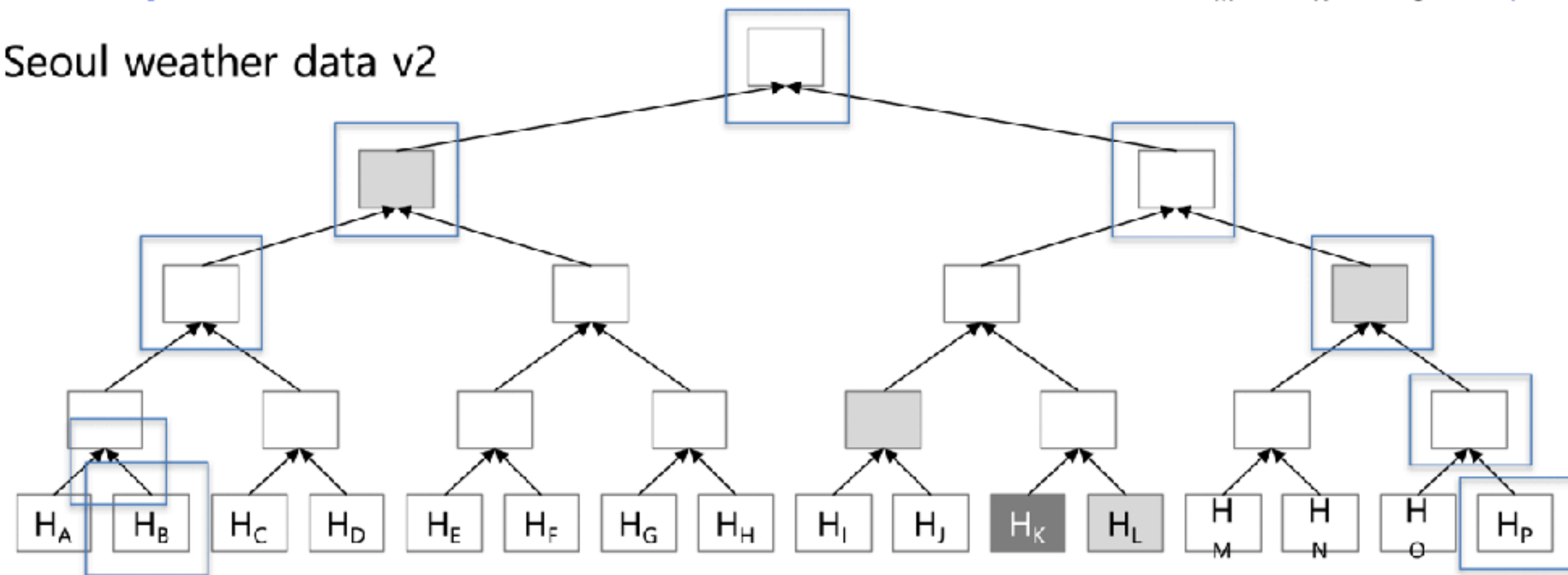
The newer version has the same depth but changes a few segments.

Seoul weather data v1



Scenario #3

Seoul weather data v2

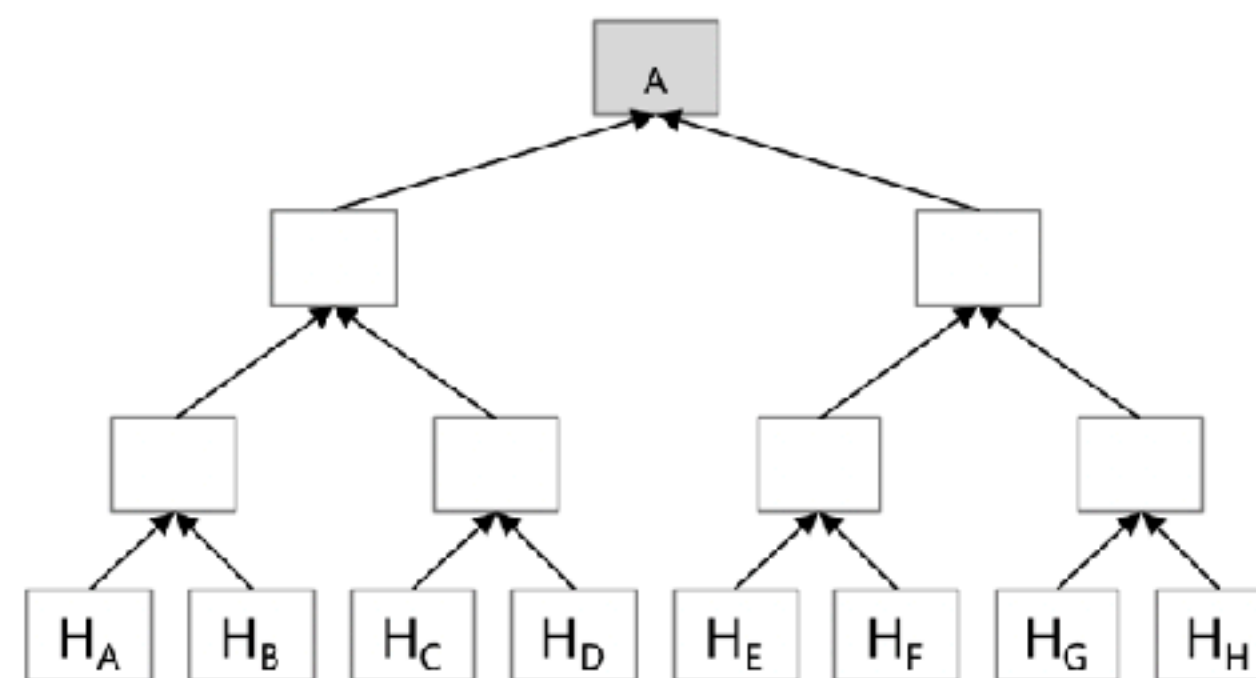




시나리오 #4

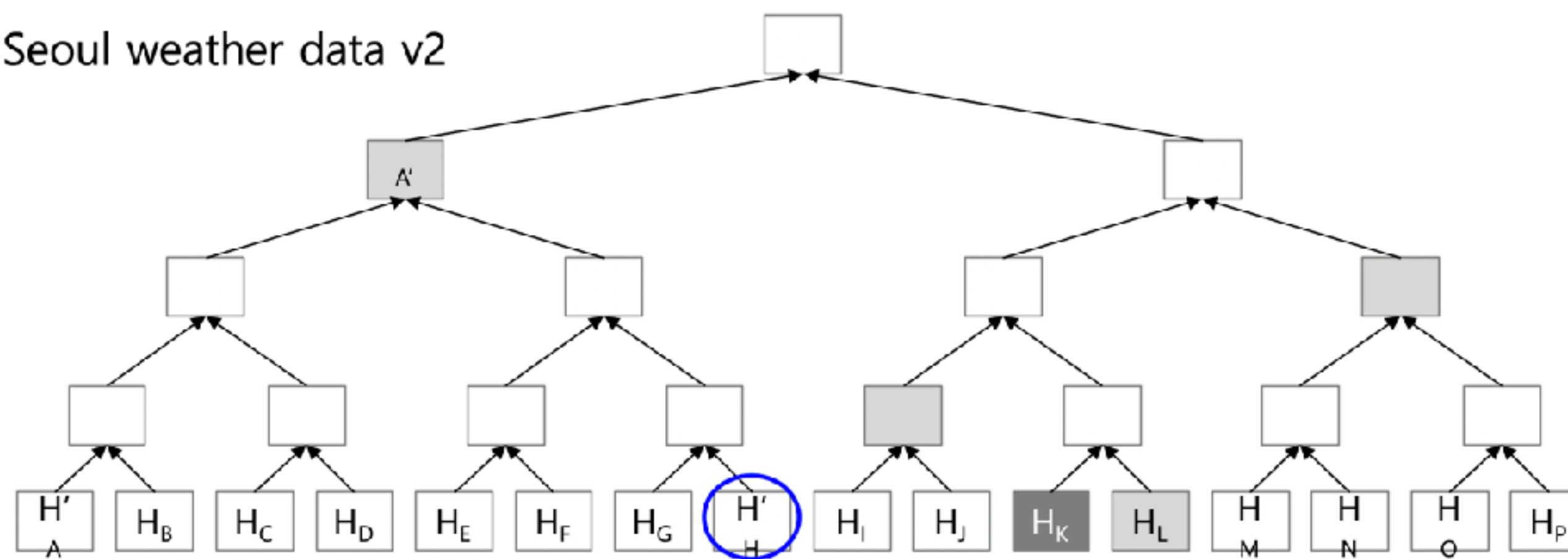
The newer version has different depth and changes segments too.

Seoul weather data v1



Scenario #4

Seoul weather data v2



일정

Schedule 2020'

- 크게 상반기 (~7월) 및 하반기 (~11월) 구성
 - 상반기: ‘기본 기능’ 및 ‘Signature’ 파트
 - 하반기: ‘Layered Hash Tree’ 파트
- 논문 및 OpenSource 일정과 함께 조율 필요
 - 상반기 일정은 보다 빠르게 진행될 필요가 있음
 - ‘기본 기능’ 및 ‘signature’ 내용을 반영한 논문 작성 예정
 - 현재 후보 conference 의 submission 일정
 - 2020.05.22. 로 계획됨



일정: 우선 순위


- 다음과 같은 우선 순위로 진행함
 - 1) 기본 기능
 - 2) Signature (blake, hash-chain)
 - 3) Layered Hash Tree
- 상세 개발 일정
 - 상호 일정 조율 예정



일정: paper 일정 참고

- acm-icn 2020 일정 (<http://conferences.sigcomm.org/acm-icn/2020/>)
 - May 15, 2020: paper registration
 - May 22, 2020: paper submission
 - Aug 1, 2020: acceptance noti
 - Sep 1, 2020: camera ready
 - Sep 28, 2020: conference
- ‘COVID 19’ 로 인한 일정 조율 가능성 있음
 - 일정 조율과 상관없이 현재 정해진 일정을 기준으로 함

News

March 22, 2020	Please note that regarding the COVID-19 virus and its consequences for the conference organization, we will follow the procedures and advises of ACM SIGCOMM.
March 17, 2020	Bruce Maggs is confirmed as the keynote speaker at ACM ICN 2020 .
March 1, 2020	Call for papers to ACM ICN 2020 Information-Centric Economic, Societal and Governance Workshop is up
Jan 17, 2020	Call for papers is up
Dec 5, 2019	Web site is up
Older News 	

Important Dates

May 15, 2020	Paper Registration Deadline (Long and Short)
May 22, 2020	Paper Submission Deadline (Long and Short)
August 1, 2020	Acceptance Notification
September 1, 2020	Camera Ready Due
September 28-30, 2020	Conference

THANKS

감사합니다