

# Initial Implementation of Interest Digest in NFD

Junxiao Shi (shijunxiao@email.arizona.edu)

## Motivation

When a Data packet arrives at NFD, current forwarding protocol requires finding all PIT entries that can be satisfied by this Data, which has significant computation cost. Interest digest can be used to improve forwarding performance: the Data packet carries a digest of the Interest it can satisfy, and the PIT is indexed by this digest, which allows NFD to quickly find the PIT entry.

## Contribution to NDN

This project starts the implementation of Interest digest in NFD forwarding plane, as a starting point to further study its impact on protocol semantics and forwarding performance <sup>1</sup>.

## Tasks

**Interest digest computation and header field** Implement a function to compute the SHA1 digest <sup>2</sup> over Name+Selectors+Link fields. Define an NDNLPv2 header field to carry the digest.

**Consumer changes** In `ndn-cxx Face::expressInterest` API, compute the Interest digest and add it onto every Interest.

**Producer API changes** In `ndn-cxx Face::put` API, require the Interest to be passed together with the Data/Nack, so that every Data/Nack is tagged with the Interest digest. Modify `ndnpingserver` (and other producer applications as necessary) to use the new API.

**Index PIT with Interest digest** Take PIT off the NameTree <sup>3</sup>. Change PIT structure: Interest digest is the only index. Change PIT insertion and lookup algorithm to support the new structure.

**Data/Nack forwarding changes** Simplify NFD incoming Data pipeline and incoming Nack pipeline because they will find at most one PIT entry.

## Required Knowledge for Participants

- C++11
- knowledge about NFD forwarding pipelines
- a computer capable of running 3 NFD instances via Mininet, Mini-NDN, or virtual machines; OR, access to Open Network Laboratory (ONL)

## Expected Outcome

**forwarding benchmark** Run a benchmark to demonstrate the performance improvement from digest-based PIT matching.

<sup>1</sup>While Interest digest is proposed as a solution to prevent cache poisoning attacks with Link objects, this project focuses on its performance benefit, and does not study cache poisoning aspect.

<sup>2</sup>While choice of digest algorithm is worth studying, it's not the focus of this project.

<sup>3</sup>Whether Interest digest could be optional is still worth further studying, but in this project we will require Interest digest on every packet and not support name-based lookups at all, to demonstrate the maximum performance potential.