ndnSIM - Task #2370

Implement emulation of ndn-cxx `ndn::Face`

01/11/2015 06:31 PM - Alex Afanasyev

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Spyros Mastorakis	**% Done:**	100%
Category:	apps	**Estimated time:**	0.00 hour
Target version:	2.1		

Description

Related issues:

Blocks ndnSIM - Task #3119: Add (at least partial) unit-test coverage for ndn...	**Closed**	**08/20/2015**

History

#1 - 03/22/2015 11:34 AM - Junxiao Shi

Connecting a nfd::Face to a ndn::util::DummyClientFace is easy.
Example: topology-tester TopologyAppLink

I'm attempting something similar in my experiments.
I feel the difficult part is:

ndn::Face is designed to be asynchronous and requires io_service, but ns-3 doesn't run io_service as its main loop.

One solution is to invoke face.processEvents(time::milliseconds(-1)) after *every* operation (including but not limited to expressInterest put setInterestFilter), which requires extensive changes in applications.

Another solution is to use a global io_service, and schedule a periodical event in ns-3 to poll it; but the frequency of polling would be a trade-off between accuracy and wallclock performance.

#2 - 06/09/2015 11:21 AM - Anil Jangam

In attempt to understand this, I did some investigations. Below are my questions and findings:

- In a real NLSR deployment, each host (router) running NLSR runs its NFD instance. Faces are managed by the application and network admin (nfdc command I mean).
    - In ndnSIM setup, does every node has its simulated NFD instance or the entire simulator has just one for all nodes in the simulation setup? I presume ndnSIM contains just has one instance of simulated NFD.
- Interest messages are dispatched to the application and to NFDs depending on the registered prefixes in the FIB by application and by the forwarder.
    - We need an equivalent of the above in NFD in ndnSIM and IMO it need not be asynchronous in nature unlike in real NFD case, correct?
    - As per an update in redmine task #2370, do we really need to monitor the activity on face in simulated NFD? Can't it just dispatch the Interest messages just looking at the registered prefix?
    - This dispatched Interest event then eventually be handled in ndnSIM's event handler.
    - I might be oversimplifying this. Please provide your comments.
- In 2.0 version, when we are creating the links between the nodes, does it actually creates the faces on NFD?
- In NLSR code, face management is done by util::FaceController. This is the class takes care of creating and destroying the faces.
    - In ndnSIM NFD, IMO, there is an equivalent class called FaceManager. This internally uses the ProtocolFactory to createFace. However, I did not find any destroyFace API.
    - Please comment if FaceManager is the class we need to enhance to support the handling of Interest messages in ndnSIM.
    - Please comment on when NLSR application in ndnSIM registers any prefix, will it be handled by FaceManager and registered into the FIB of simulated NFD.
    - As mentioned above, please comment if faces created by creating links between the nodes (based topology) are also handled by the FaceManager.

I am not sure how much time this ndnSIM face implementation would take. If there is a workaround to this, I can proceed with the further testing of nlsrsim.

Let me know your feedback.

#3 - 06/09/2015 11:22 AM - Anil Jangam

Did some further fact finding.. Here are my observations.

- In helper/ndn-stack-helper.cpp, I see that the NDN stack (L3Protocol) is installed on each NDN node.
- In the same helper class, I see that all the faces are added to the FaceContainer.
- In the same helper class, I see a FibManager, which belongs to the NFD package in the simulator. This is used to register the name prefix in the FIB and associate the face to handle the Interest/Data packet.

Is there any relationship between the faces in the FaceContainer and that of faces to be created on NFD? If I am correct, the FaceContainer contains the faces between the application and NFD. And the links between the nodes are the faces connecting the nodes.

Do we have to write a FaceManager (similar to the FibManager) for the management of the faces on the NFD. This will then replace the Link based logic connecting the nodes. Kindly please comment.

#4 - 08/07/2015 12:47 PM - Spyros Mastorakis

- Status changed from New to In Progress

- Assignee set to Spyros Mastorakis

#5 - 08/14/2015 05:12 PM - Spyros Mastorakis

- % Done changed from 0 to 40

#6 - 08/17/2015 02:15 PM - Spyros Mastorakis

- Status changed from In Progress to Feedback

- % Done changed from 40 to 90

@Alex, it works now. I will push the updated code soon to the repo. Could you please take a look and let me know if you like the structure? Instead of deleting what I think we do not need in ndn-cxx Face, I have commented the lines out.

#7 - 08/21/2015 06:16 PM - Alex Afanasyev

- Blocks Task #3119: Add (at least partial) unit-test coverage for ndn-cxx Face added

#8 - 08/21/2015 06:49 PM - Alex Afanasyev

- Status changed from Feedback to Code review

- % Done changed from 90 to 100

#9 - 09/04/2015 12:28 AM - Alex Afanasyev

- Status changed from Code review to Closed