1 Introduction
This document defines LINK object and its usage.

2 LINK Object
2.1 LINK for Redirect
By its name, a LINK object binds together two names (a name can be a prefix). In a way, this is somewhat similar to the “symbolic link” in UNIX file system.

We define LINK as a sub-type of NDN DATA packet (Figure 1). In this DATA packet,

\[ \text{Name: } /\text{ndnsim} \]
\[ \text{MetaInfo:} \]
\[ \text{ContentType: } \text{LINK} \]
\[ \text{FreshnessPeriod: (optional)} \]
\[ \text{Content:} \]
\[ \text{Name: } /\text{att} \]

Figure 1: Link

- **Name**: Name\(_A\)
- **MetaInfo**:
  - **ContentType**: LINK
  - **FreshnessPeriod**: optional, producer defined period of LINK object freshness
- **Content**: Name\(_B\)
- **Signature**: by a key that authorized to sign Data in namespace Name\(_A\). For example, in the hierarchical trust model, this is Name\(_A\) or Name\(_A\)’s parent key, confirming that Name\(_A\) has delegated a sub-namespace to Name\(_B\)

This LINK object binds together the names A and B.

LINK can be used to serve two purposes: redirect (like HTTP redirect), or encapsulation.

“Redirect” means that when a consumer sends an Interest with name Name\(_A\), the reply Data packet contains a LINK which carries another name Name\(_B\), indicating that the produce of Name\(_A\) suggests the consumer to send the Interest using name B to request the desired data. Here Name\(_A\) and Name\(_B\) can be either a prefix or an exact name.
2.2 LINK with Delegation

In many cases, it is necessary to ensure not only that an entity A has linked some name prefix from namespace A to namespace B, but also that B has authorized A to perform such actions. For this purpose we define Delegation link object (Figure 2), which contains the following information:

- **Name**: Name$_A$

- **MetaInfo**:
  - **ContentType**: DELEGATION
  - **FreshnessPeriod**: (optional)

- **Content**: LINK object “redirecting” namespace B to a prefix of name A. This LINK object ensures that owner of namespace B actually authorized owner of the namespace A to perform redirections.

- **Signature**: by a key that authorized to sign Data in namespace Name$_A$. For example, in the hierarchical trust model, this is Name$_A$ or Name$_A$ ’s parent key, confirming that Name$_A$ has delegated a sub-namespace to Name$_B$

2.3 LINK for Data Retrieval (Encapsulation)

“Encapsulation” is needed when consumer wants to fetch Data whose prefix P$_u$ is not announced to the routing system. To be able to fetch this Data, there must be a routable prefix P$_r$ so that a certain way of concatenating P$_r$|P$_u$ can get the desired data back.

More specifically, to retrieve a Data packet with a name N$_u$ that is not directly routable towards producer, a consumer needs to perform the following steps:

- perform a resolution N$_i$ ⇒ P$_r$ (e.g., using NDNS), which should return a LINK object defined in the previous section. This LINK is binding an unroutable prefix of N$_i$ (this prefix can/will be discovered during resolution process) to routable prefix P$_r$.

- construct the name N$_r$ that contains P$_r$ and N$_u$ (e.g., concatenation or concatenation with marker delimiter1), and sends Interest packet

The retrieved Data packet should conform to the following format (Figure ??):

- **Name**: Name$_{outer}$

- **MetaInfo**:

1The delimiter may be necessary to disambiguate prefixes during forwarding
2.3.1 Details to Work Out

Details on how Nameouter is constructed is TBD.
Figure 4: Linked Data (option 2)

Figure 5: Linked Data (option 3)