

INTERNET-DRAFT  
Intended Status: Informational  
Expires: August 21, 2013

Rong Wang  
Chunfeng Yao  
Zhefeng Yan  
Huawei  
February 17, 2013

Container Resolution System in ICN  
draft-ietf-icnrg-icn-container-name-system-00

## Abstract

We illustrate the architecture and mechanism of container resolution system, and to the protocol of resolving containers in ICN. The fundamental characters are: 1) Use Interest and Data packet as the communication primitives, 2) Every routing node has FIB, CS, PIT tables to do packet routing and caching. Current NDN/CCN architecture does not require a resolution system. The routing is totally based on prefixes of content names, which causes scalability and mobility issues described in [Cisco-Name]. This draft addresses the aforementioned issues.

## Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at  
<http://www.ietf.org/lid-abstracts.html>

The list of Internet-Draft Shadow Directories can be accessed at  
<http://www.ietf.org/shadow.html>

## Copyright and License Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1	Container Name System . . . . .	3
1.1	Introduction . . . . .	3
1.2	System Architecture . . . . .	3
1.3	System Components of CNS . . . . .	4
2	Container Name Scheme . . . . .	5
2.1	Content Name Composition . . . . .	5
2.2	Container name Registration . . . . .	5
2.3	Unregister Container name . . . . .	7
2.4	Resolve Container name . . . . .	7
2.5	Packet flow in network . . . . .	9
3	Implementation of CNS . . . . .	11
4	Execution Engine Process Logic . . . . .	13
5	IANA Considerations . . . . .	14
6	Conclusions . . . . .	14
7	References . . . . .	14
7.1	Normative References . . . . .	14
7.2	Informative References . . . . .	14
	Authors' Addresses . . . . .	16

## 1 Container Name System

1.1 Introduction This draft is the companion to [Huawei-name] and illustrates the container resolution system (hereinafter referred to as NRS) . The NRS assists the routing requirement of the container name, which is the extension of conventional Information Centric Networking (hereinafter referred to as ICN) naming.

### 1.2. System Architecture

An NRS is composed by a number of distributed container name systems (hereinafter referred to as CNS) deployed in an ICN network. CNS maintains the mapping from containers to its access containers. The concept of container, container name, container set , container assisted routing, resolvable container name, and access container are described in [Huawei-name]. The following figure illustrates the high level CNS architecture.

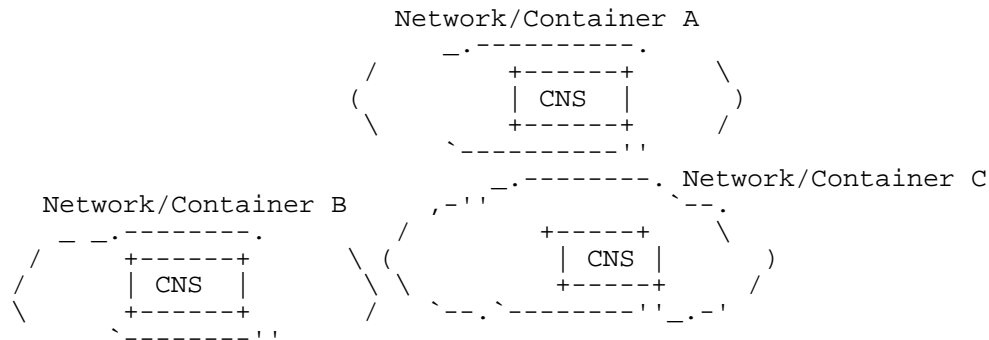


Figure 1

For each container name, there is an authoritative CNS (hereinafter referred to as A-CNS) acting as its final name resolution provider and usually residing within the container (network). One container can have an A-CNS for multiple containers. Other CNS can identify the A-CNS of its container via the container name, e.g., with one of the following methods:

- Use category tree hierarchy similar to current DNS, and configure an A-CNS for each container.

- Leverage container assisted ICN routing mechanism to identify the A-CNS of the container.

Since containers are part of the hierarchical tree or DAG of a content name, they have to know their predecessor and successor(s) in

the tree or DAG [Huawei-name]. Thus CNS within a container has the similar knowledge of its predecessor and successor(s). If a CNS of a container cannot locate the A-CNS of its container, it asks the help from its upper level CNS.

Registration and unregistration operation of a container has to be done on its A-CNS. Other CNS, for example the CNS for the local access container, proxies the registration and unregistration requests to the A-CNS. In other words, the final resolution result of a container name must come from its A-CNS. For any resolution request, if an intermediate CNS has the result cached, then it can reply directly, otherwise this CNS has to forward the request to its A-CNS, and caches the resolution result according to the predefined policy.

### 1.3. System Components of CNS

CNS has 3 components, DataBase (hereinafter referred to as DB), Execution Engine, and Network interfaces.

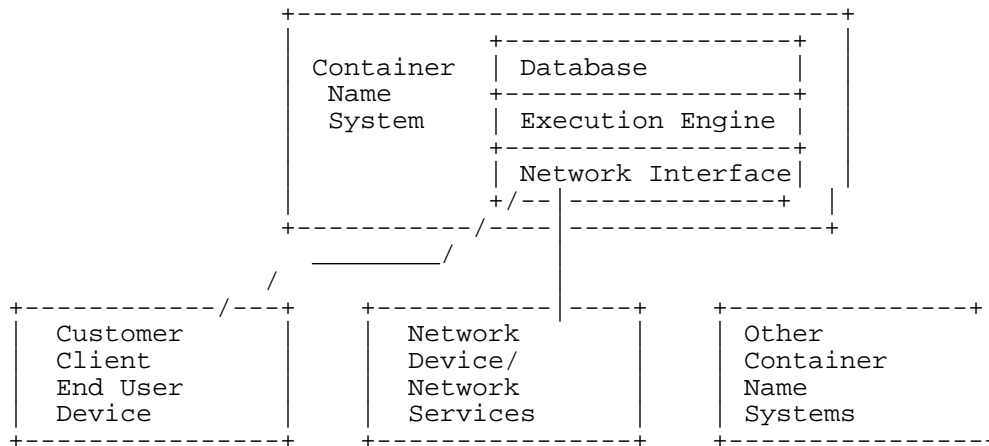


Figure 2

Database stores resolution related data. Implementation details can leverage current DB technologies or file systems.

Execution engine is responsible for processing resolution related logic. It analyzes packet from network interface, operates on DB, and sends operation result out of Network interfaces.

Network interfaces sends/receives ICN packets according to ICN protocols. It receives Interest packet from client, end device, network node, service or other CNS, and sends packet to the execution

engine for further processing. It forwards Data packet by following the instruction from execution engine.

## 2 Container Name Scheme

### 2.1 Content Name Composition

We propose to reserve a few keywords in ICN for the container name resolution operations. They are RESOLVE\REGISTER\UNREGISTER, representing 3 primitive operations: as inquiry, registration and unregistration, respectively. Keywords are configured automatically or manually during ICN joining into the network.

Once a CNS with resolution capability joins into ICN, it publishes aforementioned keywords(RESOLVE\ REGISTER \ UNREGISTER) according to the ICN rules. Any ICN node who receives the published packet will add table entries associated with those keywords to its FIB. All those entries point to the CNS.

We then prefix resolution keyword to the content ID required resolution to be the new content ID of ICN packet (Interest/Data). Since each node in ICN use LPM to lookup FIB for forwarding, any match of the keyword will route the resolution request to the aforementioned CNS.

We can also use container assisted resolution to route packet to a CNS. In this case, CNS publishes the name of the container that it resides in [super container] to ICN according to ICN rules. Any node in ICN will add routes accordingly. We can use the routing protocol described in draft [Huawei-name] to route interest to CNS by adding super container name as the suffix to aforementioned Content ID.

For example, CNS has published that it resides in the super container named RESOLVER to the ICN. The interest to inquiry "chinamobile/johndoe" can be composed as "RESOLVE/chinamobile/johndoe|RESOLVER".

### 2.2. Container name Registration

The registration operation utilizes ICN protocol's Interest packet to carry the request, and uses Data packet to carry the reply (resolution result).

In details, information of container A registration includes:  
"replace (Yes/No){access provider Container B/Container set C with its properties (attributes)}", for example: "( replace=yes;){cn/gd/sz ; TTL =100 | hostsrv.com; resolvable=yes; TTL =5000; }". Where:  
-Value of Replace field can be Yes or No (True/False?) If container A

exists in CNS,  
 --If Replace value is No: Insert the new resolution result to the front of result list. If container B or container set C has already existed in CNS, then delete the old result.  
 --If replace value is Yes: Directly replace the existing resolution result with the new one.  
 --By default, "replace=no", even with the absence of REPLACE field.  
 --Example: "Register /chinamobile/johndoe" -> "{cn/gd/sz ; TTL =100 | hostsrv.com; resolvable=yes; TTL =5000; }" For now, original resolution result of "chinamobile/johndoe" will be invalid and will be replaced by "cn/gd/sz" and "hostsrv.com"  
 -Resolvable value (Yes/No): Whether the resolution result (container B/container set C) can be resolved again. The operation of recursive resolution can be seen as the expansion of resolution results. The expansion resolution can be done when writing to Database, or when resolution system idling, or when receiving a resolution request.  
 -(Optional) TTL (Unit Second): Time of the resolution result can be cached. The result will not be cached if this value is 0.

Registration information can be put into Selector field. A new property, RESOLVE is added to represent name solution (see Figure 3) to hold register and unregister information.

In the Data packet, it also carries the success or failure information of the registration operation. In this reply Data packet, the "signed info" field has to set as "no caching" (This is implementation dependent), this setting ensures upcoming Interest requests will not be replied by the intermediaries but the CNS.

Aforementioned content name register and unregister information , it can be put directly into content name field rather than in Selector field. The whole content name will be composed of content ID and content name register/unregister information. As illustrated in Figure 4.

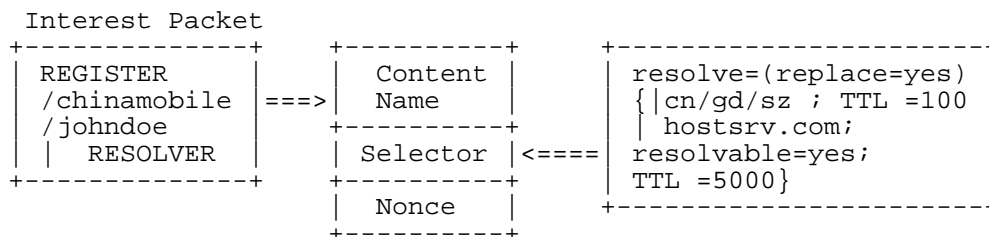


Figure 3

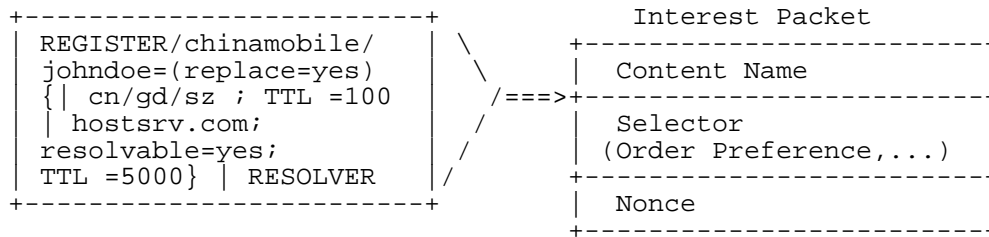


Figure 4

### 2.3 Unregister Container name

The purpose of this operation is to unregister the access container B/container set C for container A. In other words, container B/container set C is removed or voided from container A's resolution result list. This operation piggybacks on ICN protocol's Interest packet to carry the request, and leverages Data packet to carry the reply.

In details, the unregistration information of a container name looks like: "{access provider container B/container set C}". For example, "{cn/gd/sz ; | hostsrv.com;}". This information is in the new added properties in Selector field similar to that of registration operation.

-If container A does not exist in CNS, or there is no aforementioned container B/container set C as the resolution result of container A, then CNS replies failure status.

-If there are resolution results, remove them from CNS.

-If all resolution results are removed successfully, then reply "total success", otherwise, reply "partial success" with failed portions and reasons. The common reason is "non-exist".

-If resolution result of container A does not exist, then remove container A from CNS. The operation result will be carried back by Data Packet similar to that of registration. The Data Packet will be set as uncacheable.

### 2.4. Resolve Container name

The purpose of this operation is to obtain container B/container set C which provides access service for container A. This operation utilizes ICN protocol Interest Packet to carry the request, and Data packet to carry the resolution result.

In details, the carried packet has to include content name A. For example "chinamobile/johndoe"

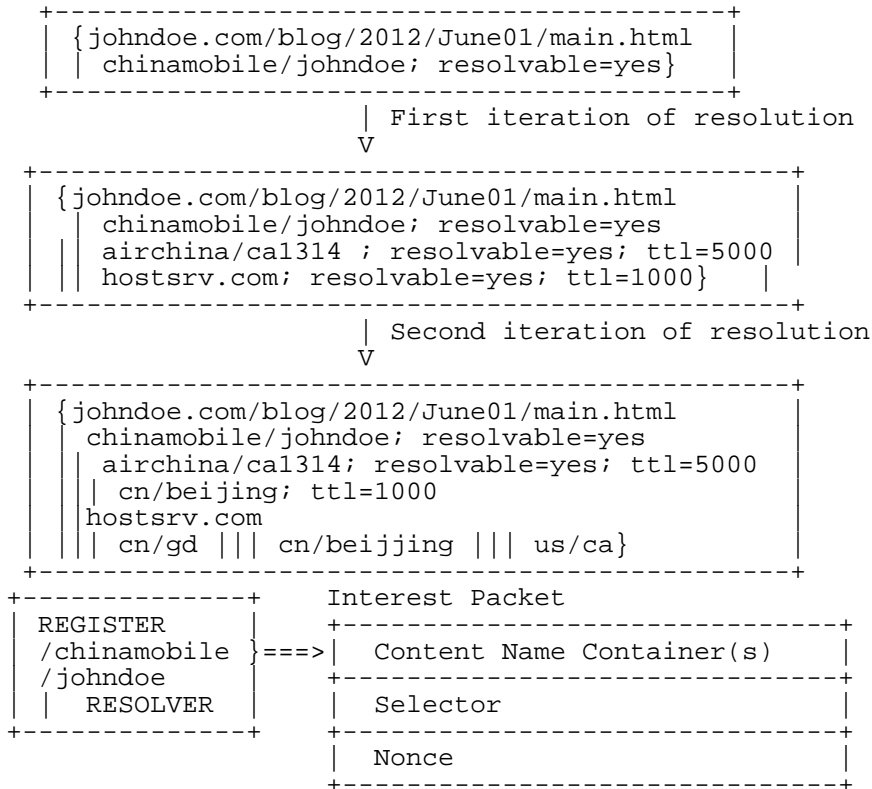
The resolution result is carried back by Data packet in the Data field, for example, "airchina/ca1314; resolvable=yes;ttl=5000; ||| cn/beijing; ttl=1000;|||hostsrv.com ||| cn/gd ||| cn/beijing ||| us/ca", CNS will do the following:

-If content name A does not exist in the CNS, then replies "failed" or NULL.

-Return the resolution result of container A in order. If the result container name has the "resolvable=yes" property, then it can be further resolved. The resolution results can be inserted in the original result by following the method described in [Huawei-name]

-If the result has TTL property, the caching timer at Signed Info field of Data packet (implementation details may be different, for example, in CCN, it is FreshnessSecond, in NDN it is staeltime) will be set as the minimum of TTL (in previous example, it is 1000)

-The intermediaries ICN nodes can cache the resolution results in its Content Store (hereinafter referred to as CS). Cache Time can be processed according to ICN rules. Request packet is shown in Figure 5.





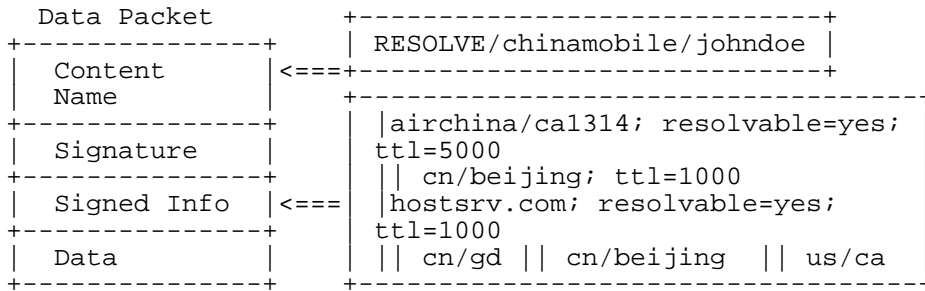


Figure 5

2.5. Packet flow in network

Packet flow of a client/request node requesting container registration/unregistration is shown in Figure 6.

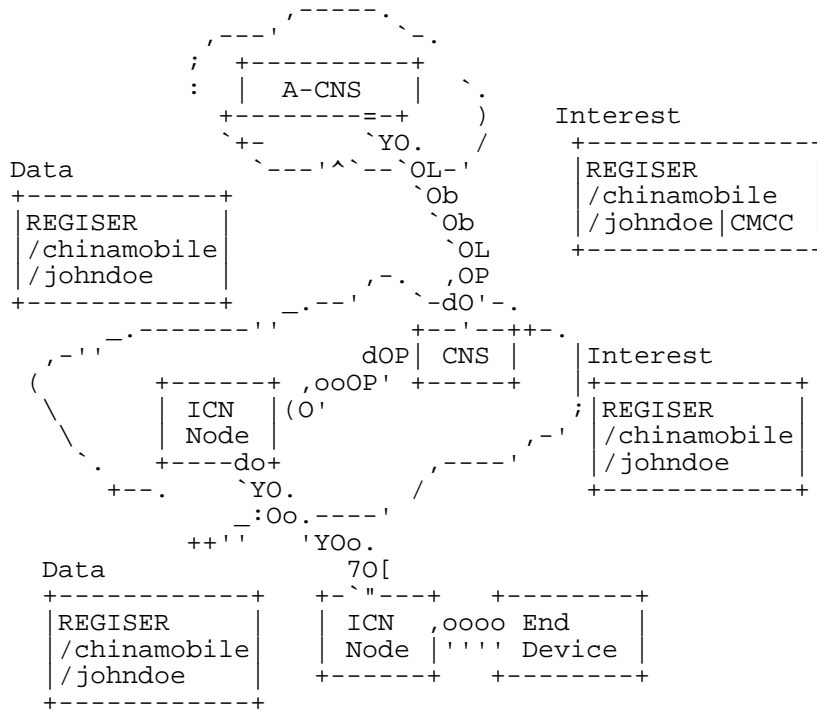


Figure 6 container registration/unregistration  
 -Utilize Interest Packet to request name resolution to ICN.  
 -Local CNS processes the Interest Packet first.

- If the local CNS is the aforementioned A-CNS of this container, then registration will be done. A Data packet with result will be sent back.
- If not, the Interest packet will be routed out to the next level CNS or A-CNS (for example, CMCC) of container A, per the preset rules, until the registration/unregistration operation can be done on A-CNS.
- The intermediaries CNS will forward the Data packet to the original requester once it receives the Data packet from next level CNS or A-CNS.
- Data packet of Register/Unregister does not allow to cache by intermediaries. This can be done by set "no-caching" in Data packet in ICN protocol.

Packet flow of a client/a node requesting a resolution of container is shown in Figure 7.

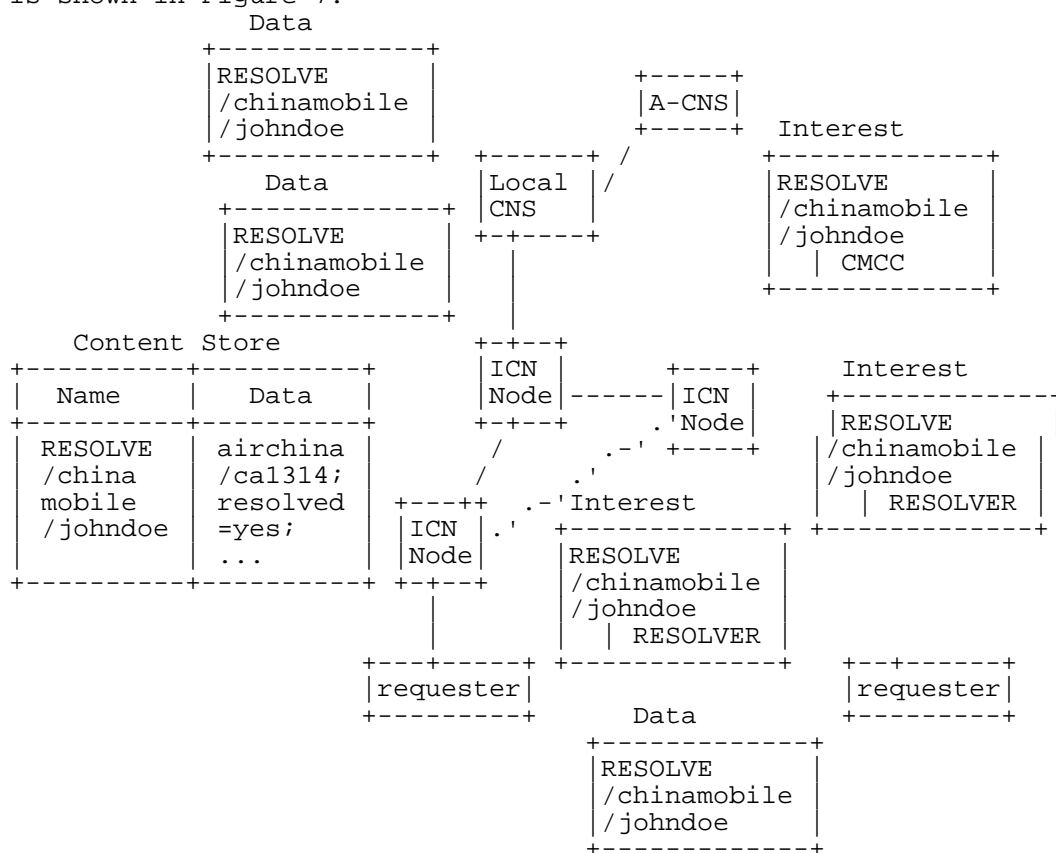


Figure 7 resolution of container

- Utilize Interest Packet to request name resolution to ICN.
- Local CNS processes the Interest Packet first.
- If local CNS can resolve the request, it returns a Data packet with resolution result.
- If it cannot be resolved, then appends the name of next level CNS or the name of A-CNS (for example, CMCC) of container A by following the preset rules. Route the Interest packet accordingly.
- After gets Data packet back from next level CNS or A-CNS, then it inserts the resolution result into its Database. It generates a new Data Packet.
- The intermediaries passed by Data packet will forward the Data packet as well as cache it in its CS table, according to the content forwarding rule.
- If the same request comes into this intermediary, then Data packet will be fetched from cache and return back to requester, according to the content forwarding rule.

### 3 Implementation of CNS

Aforementioned CNS is based on ICN node (support ICN protocol at lower level). It is responsible for name registration and resolution. It can be part of a resolution system together with other CNS'.

Any end device/network device/service can register its access container/container set in the CNS to enable other devices and services to inquiry them. Meanwhile, it has the capability to work seamlessly with other CNS.

As illustrated by Figure 8, John Doe applied a personal domain name "chinamobile/johndoe", China Mobile provides access service for him. Current access location is Shenzhen, Guangdong. Thus he registered the container as "cn/gd/sz"; the same user applied a third party (Hostsrv) to provide him with service. The name of the specific access container of the service can be obtained via resolution lookup (For example, Hostsrv provides 3 DCs for John Doe at "cn/gd", "cn/Beijing", and "us/ca"). Thus the container is registered as "hostsrv.com resolvable=yes". Once ICN receives Interest packet with name "chinamobile/johndoe", it sends resolution request to CNS. After two iterations of resolutions, NRS returns the resolution results.

<pre> +-----+   Register     /chinamobile     /johndoe --&gt;   +-----+ </pre>	<pre> Interest +-----+   Content Name =     {johndoe.com/blog     /2012/June01     /main.html      chinamobile/johndoe;   +-----+ </pre>
--	--

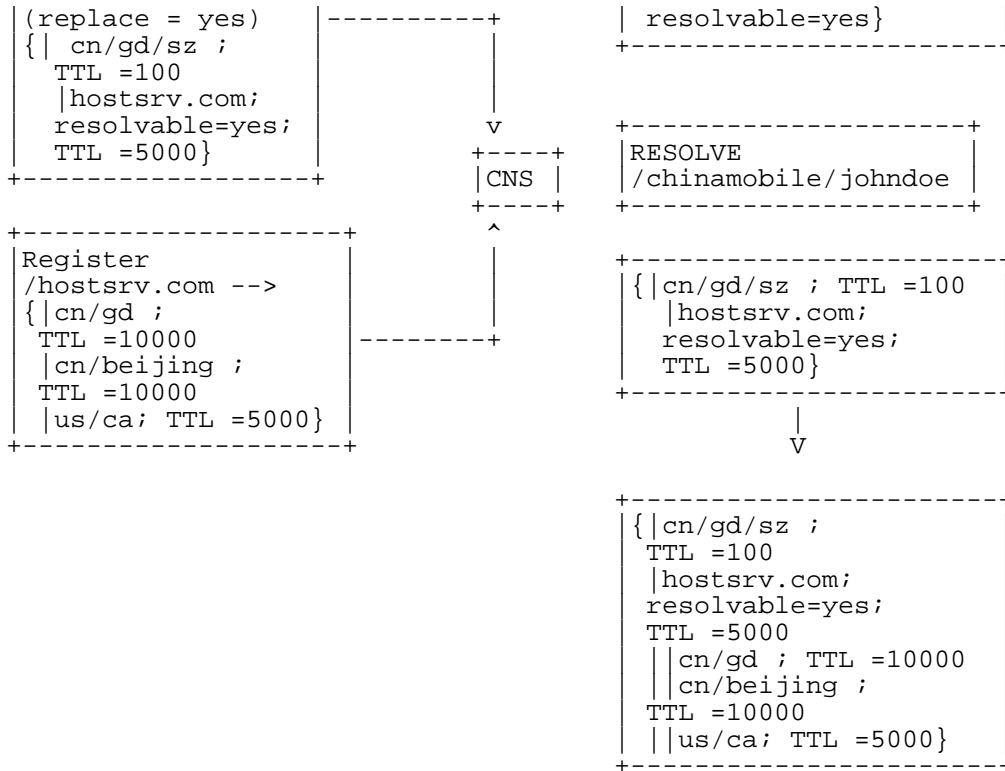
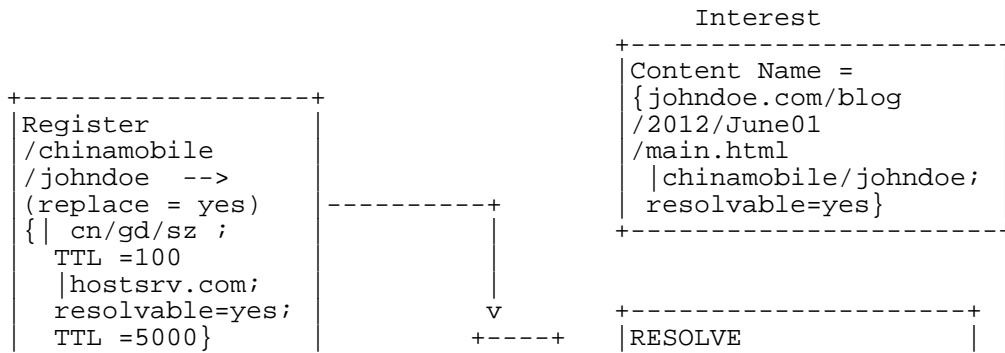


Figure 8

Another implementation method is to return the direct resolution result. If client wants to resolve the resolvable container, then it can send another resolution request. Figure 9 illustrates the procedure.



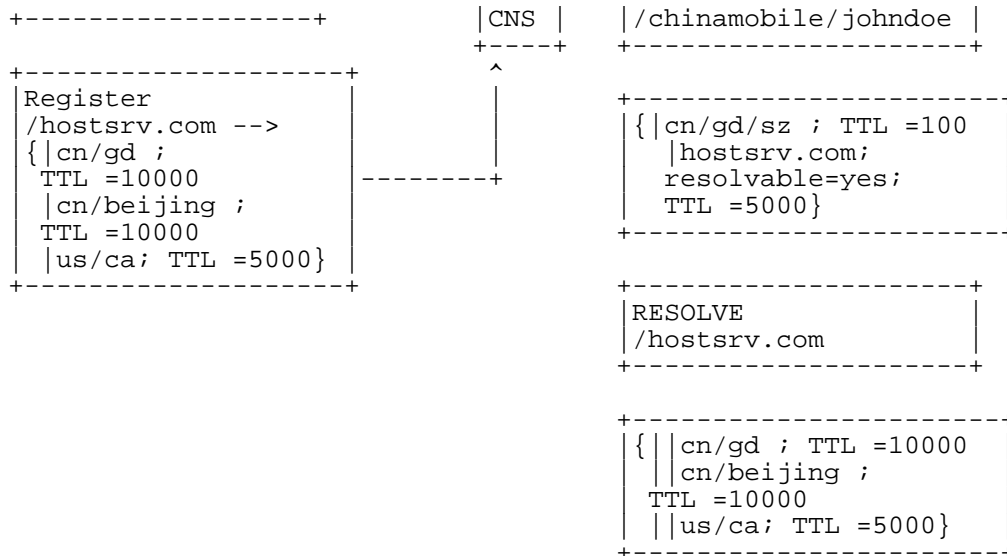


Figure 9

#### 4 Execution Engine Process Logic

##### Detailed steps:

1. Is the receiving packet Interest packet or Data packet? If it is interest packet, go to step 2, otherwise, go to step 10.
2. Check operation type. If it is "register/unregister" type, then go to step 3. If it is "resolve" type, then go to step 8.
3. Use this node as A-CNS? Yes, go to step 4, otherwise go to step 9.
4. If it is "register" type, go to step 5, otherwise remove container names identified by Interest packet from the result list, go to step 12.
5. If container name about to register exists, go to step 6, otherwise, add new container name and associated resolution results to Table. Generate Data packet with successful status, go to step 7.
6.
  - i. If "replace=true", then replace the existing resolution results with the ones carried from interest packet, generate Data packet with successful status, go to step 7.
  - ii. If "replace=false" or no replace property, then insert resolution results from interest packet to the front of the container list. If container name in interest packet is the same as of old resolution results, remove original associated table entry. Generate Data packet with successful status, go to step 7.
7. (This optional step can be skipped and directly go to step 12). Traverse resolution results; index all resolvable containers. If a

resolvable container can't be resolved locally, then route Interest packet to the next level CNS or A-CNS of the container name. Wait for result carried back by Data packet. If the result is NULL, this container name cannot be resolved. Return failure status in Data packet, otherwise, create index to the next level resolution. Generate a complete Data packet with resolution results from all iterations, go to step 12.

8.

i. Does container name exist in the table? If not, then go to step 9. If yes, then traverse all resolution results, insert generated Data packet. Go to step 12 after all results have been checked.

ii. (Optional Step) If some of the results are resolvable ("resolvable=yes"), then resolve the container recursively by using this step. If the same result has been resolved (the same container name has been packaged into Data Packet), then ignore the result, and keep the traverse the next item (container name).

9. Append container name of next level CNS or A-CNS to the content ID in Interest Packet per preset policy. If there is a container name associated to the node, remove it. Generate a new Interest packet and route it out, and waiting for the returning Data packet.

10. If the reply Data packet is for register/unregister request, forward Data packet to the original requester, go to step 12. If the reply Data packet is for resolution request, go to 11.

11. Create table entry per the content ID (container name in this context) and resolution result within the Data packet, If the same container name exists, replace or remove it from the table per the preset rules. Generate Data packet based on local table results. If resolution result is NULL then no table entry is created. If this resolution is triggered by previous REGISTER resolution, go to step 7, otherwise, it is caused by RESOLVE resolution, go to step 6.

12. Call network interface module to forward Interest or Data packet.

## 5 IANA Considerations

No IANA consideration for this draft.

## 6 Conclusions

## 7 References

### 7.1 Normative References

### 7.2 Informative References

[Cisco-Name]

NDN and IP Routing, Can it Scale?  
<http://trac.tools.ietf.org/group/irtf/trac/raw-attachment/wiki/icnrg/IRTF%20->

%20CCN%20And%20IP%20Routing%20-%202012.pdf

[CCN] V. Jacobson et al. Networking named content. Proceedings of the 5th ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT 2009). NY: ACM; 2009; 1-12.

[ICN-name] Ali Ghodsi et al. Naming in Content-Oriented Architectures. In ACM ICN workshop, 2011.

[SCN] D. Smetters and V. Jacobson. Securing Network Content. Technical report, PARC, October 2009.

[Huawei-Name] Draft-ietf-icnrg-icn-naming-and-routing-00.txt.

Authors' Addresses

Rong Wang  
Huawei Technologies  
Huawei Building,  
BanTian, LongGang District  
Shenzhen, GuangDong 518129  
P.R.CHINA

EMail: WANG.RONG@huawei.com

Chunfeng Yao  
Huawei Technologies  
Huawei Building,  
No.156, BeiQing Rd., Haidian District  
Beijing, 100095  
P.R. CHINA

EMail: chunfengyao@huawei.com

Zhefeng Yan  
Huawei Technologies  
Huawei Building,  
BanTian, LongGang District  
Shenzhen, GuangDong 518129  
P.R.CHINA

EMail: yanzhefeng@huawei.com