# Network Resource Lookup Protocol

Version 1.0

Drafted by Shulai Zhang

Contributors: Qiaoqiao Li, Bingjue Sun.

## Specific Words Explanation

###### Resource:

In this article, it refers to the data that is provided by some services and requested by other in a certain network. There are static resources(e.g., file, constant, etc.), as well as dynamic resources(e.g., HTTP session object, outcome of a called function, etc.).

###### Resource Provider:

In this article, it refers to the system that provides some resource in a certain network. It could be a service or application written in whatever language.

###### Resource Requester:

In this article, it refers to the system that requests some resource in a certain network.

Be aware that a system could be both a resource provider and a resource requester at the same time.

###### Central Message Service:

In this article, it refers to the service responsible for transfer message between systems in a certain network. It is the Central Message Service that makes all the systems in the network can communicate with each other.

## Protocol Description

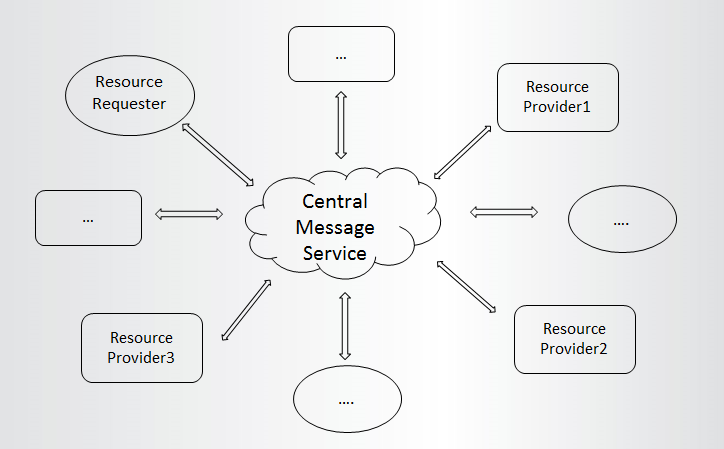


Figure1

This protocol is based on a existing message service, so that is a priority. It is usually implemented to handle tasks in multi-application environment, as well as clustered environment.

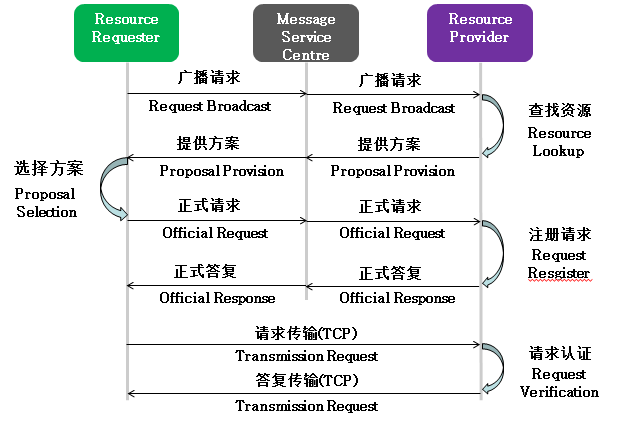


Figure 2. the Workflow View

## Protocol Details

#### Message Format

Every message is composed by two parts, message head and variables.

#### Message Head

All NRLP message should begin with a 8-byte message head.

Including:

* Protocol Name. UTF-8 encoded. 6 bytes total.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Description | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  |  |  |  |  |  |  |  |  |  |
| Byte 1 | Length MSB (0) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Byte 2 | Length LSB(4) | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Byte 3 | ‘N’ | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| Byte 4 | ‘S’ | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| Byte 5 | ‘L’ | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| Byte 6 | ‘P’ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |

* Protocol Version. 1 byte.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|  | Protocol Version | | | | | | |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

* Message Type. 1 byte.

|  |  |
| --- | --- |
| Byte Value | Message Type |
| 0 | Reserved |
| 1 | Request Broadcast |
| 2 | Proposal Provision |
| 3 | Official Request |
| 4  5 | Official Response  Transmission Request |
| 6 | Final Response |

#### Variables

###### Client ID:

Every single instance of application in this network should have a unique ID, so it makes others know who is actually speaking. It should be UTF-8 encoded in the message body.

###### Request Code:

It is sensible logic that a requester can send multiple request at the same time. So it is the requester's responsibility to keep them identical. The simple way is give each one of them a unique code. This code should be reusable when a request has done. It is a 2 bytes unsigned integer, so the maximum size is 65535.

###### Transmission Code:

Correspond to the request code, there is a transmission code in the provider side. It is generated once a transmission is confirmed. That is after official response. The implementation is the same as request code, 2 bytes unsigned integer.

###### Resource Name:

Each resource should have a unique name in this network, so it can be picked up when remote client searches with name. UTF-8 encoded.

###### Resource Type:

Every resource is marked to a single type, so when client gets the resource it can handle it separately regarding to its type. It is a 2 bytes unsigned integer.

|  |  |
| --- | --- |
| value | Resource type |
| 0 | Static Resource. When requester broadcasts a request with this code, the provider should look up the resource from all resources that have resource type between 0 to 29,999. |
| 1 | File Resource. |
| 2～9999 | Reserved. |
| 10000～29999 | Customization available. |
| 30000 | Dynamic Resource. When requester broadcasts a request with this code, the provider should look up the resource from all resources that have resource type between 30,000 to 65535. |
| 30001 | HTTP Session |
| 30002 | Java Serializable |
| 30003～39999 | Reserved. |
| 40000～65535 | Customization available. |

###### Transmission Type:

It is the way both requester and provider have agreed how to transmit the data. 1 byte long, Up to 255 transmission possibilities.

|  |  |
| --- | --- |
| value | Transmission type |
| 0 | One-shot transmission. |
| 1 | Resume-able transmission |
| 2 | Unique-maintained transmission |
| 3～99 | Reserved. |
| 100～255 | Customization available. |

###### Resource Size:

Resource size should be included in the Proposal Provision. This part could be a quite large number. It is encoded using Multi-Byte Integer algorithm.

###### Transmission Speed:

It is the maximum number of bytes that will be written out by the provider. It is a 2-byte unsigned Integer. Each unit stands for 1 kb/s.

###### Resource TCP Location:

It is the TCP location of the requesting resource. The requester will establish a tcp connection to the location for fetching the resource.

###### Remaining Length:

This value is representing the number of bytes remaining within the current message. It uses Multi-Byte Integer encoding algorithm.

###### Request Parameters:

It is representing the parameters passed to the provider in the Transmission Request message. This might be used to generate the dynamic resource that is requested, such as the return value of a method.

###### Response Code:

It is a byte representing the result of request in the Final Response. Currently, number One stands for “OK”, and number Zero stands for “Rejected”.

#### Message Details:

##### Request Broadcast:

The resource requester broadcasts his demand over the network.

Variables:

1. The requester’s client ID in this network. It should be unique in this network.
2. The request code.
3. The requested resource name.
4. The requested resource type.
5. The transmission type.

##### Proposal Provision:

The resource provider provides a proposal that defined the detail of the resource transmission, including who is providing, what the transmission type would be and the transmission speed.

Variables:

1. The request code. It should be the same as received in Request Broadcast.
2. The resource size. The total size of the resource if it is a static resource.
3. The provider’s client ID in this network. It should be unique in this network.
4. The transmission type the provider provides.
5. The transmission speed the provider provides.

##### Official Request:

After the requester selects one of proposals from all the providers, it sends an official request to the target provider and confirm with details of the transmission.

Variables:

1. The requester’s client ID.
2. The request code.
3. The resource name.
4. The resource type.
5. The transmission type.

##### Official Response:

After receiving the Official Request from the requester, the provider sends a response with a transmission code (played as a token of this transmission task) and the TCP location where to fetch the resource.

Variables:

1. The request code.
2. The provider client ID.
3. The transmission code.
4. The resource TCP location.

##### Transmission Request:

After receiving the Official Response, the requester parses the TCP location from the message and connects to the TCP location with the below variables.

Variables:

1. The transmission code.
2. The remaining length.(The byte length of the parameters.)
3. The request parameters.

##### Final Response:

After receiving the transmission request, the provider does the final verification and send the response that may or may not including the resource data according to the result of verification.

Variables:

1. The request code.
2. The result code.
3. The remaining length. (The size of the resource data)
4. The resource data in bytes.