Node Search Protocol (NSP)

Introduction:

OpenLCB requires a mechanism to allow a node to search for other nodes on the network that match a particular criteria. This criteria is based on a Key=Value system. A node that wishes to be included in a search interaction shall implement this protocol. Initially this system will be developed for Train Control but there is no reason why this can not be extended for custom manufacturer defined searches.

The selected method is to use human readable string "keys" that are assigned Values (data) where you search, add and remove keys through custom MTIs messages specific to this protocol. Internally the node may store and associate the Key and Value in anyway it is most convenient for it to do so.

There are a few ways to do this. The most obvious is to have human readable keys such as:

ID="User Definable ID"
Road Name="Rio Grande Southern"
Road Number="455"
Model="K27"
Mfg="Blackstone"
Owner="Jim Kueneman"
DCC Address="455"
DCC Protocol="long"
Comment="Post wreck cab"
Decoder="Soundtraxx"
Decoder Family="Tsunami"
Decoder SW Version; "4.1"
Motive Power="Steam"
Sound Equipped="Yes"

A second method of implementing this is to use simple keys that "map" to human readable strings. This allows more efficient access for an algorithm to find the key of interest but it adds a second level of abstraction.

0="Some ID"
1="Rio Grande Southern"
2="455"
3="K27"
4="Blackstone"
5="455"
6="long"
7="Post wreck cab"
8="Soundtraxx"
9="Tsunami"

where:
0: "ID"
1: "Road Name"
2: "Road Number"
3: "Model"
4: "Manufacturer"
5: "DCC Address"
6: "DCC Protocol"
7: "Comment"
8: "Decoder"
9: "Decoder Family"

The basis of this protocol proposal will take the more direct path as it is cleaner and simple to implement. If the hardware implementation proves this to have unacceptable performance it will be re-evaluated then.

This protocol works much like the Simple Node Identify Info where the Request/Reply is repeated until the final 0x00 null is sent/detected.

Messages

	MTI	Data Bytes
NSP Request Global	19[TBD]sss	{OP_CODE} [Data Bytes]
NSP Request Addressed	19[TBD}sss	{Dest} {OP_CODE} [Data Bytes]
NSP Reply	19[TBD]sss	{Dest}{ReplyCode} [Data Bytes]

NSP Request Global

Description	Frame Count	Byte 0 [OpCode]	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Search by Key String	1-N	0x04	[SearchOptions]	[Key String] 0x00 [Mask] 0x00					
Search Key as Integer	1-N	0x05	Range	Integer (4 bytes)	[Key String] 0x00				
Terminate Search	1	0x85							

NSP Request Addressed

Description	Frame Count	Byte 0	Byte 1	Byte 2 [OpCode]	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Add New Search Criteria as string	1-N	Destination Address		0x01	[Key String] 0x00 [Value String] 0x00				
Add New Search Criteria as integer	1-N	Destination Address		0x02	[Key String] 0x00 [4 Byte Integer]				
Delete Search Criteria	1-N	Destination Address		0x03	[Key String] 0x00				
Return Key List	1	Destination Address		0x81	[Max Result Count]	[Result Max Len]	[Offset Index]	[Key String] 0x00	
Search by Key String	1-N	Destination Address		0x04	[Max Result Len]	[SearchOptions]	[Key String] 0x00 [Mask] 0x00		
Search Key as Integer	1-N	Destination Address		0x05	[Max Result Len]	Range	Integer (4 Bytes)	[Key String] 0x00	

Search Options

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Description	Reserved	Reserved	Reserved	Reserved	Single Match/Multiple Matches	Ends with	Contains	Starts with

NSP Reply

Description	Frame Count	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
NSP Reply	1-N	[Reply Code] (see below)	Data 0, N	Data 1, N	Data 2, N	Data 3, N	Data 4, N	Data 5, N	Data 6, N

Reply Code

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Description	Reserved	Reserved	Reserved	Result Longer than passed Max Length	Complete /Partial Result	OpCode Not Supported	Out of Memory	Success/Fail

SUCCESS_COMPLETE_RESULT = 0x00
SUCCESS_PARTIAL_RESULT = 0x08
SUCCESS_COMPLETE_RESULT_BUFFER_TO_SHORT = 0x10 ?????
SUCCESS_PARTIAL_RESULT_BUFFER_TO_SHORT = 0x18
FAIL_UNKNOWN = 0x01
FAIL_OUT_OF_MEMORY = 0x03
FAIL_NOT_SUPPORTED = 0x05

Implementations:

- Node A sends a global NSP Request with a Search Criteria
- Node B find a *Value* matching the Search Criteria and sends a single NSP Reply
  - If the matching *Value* fits in the first frame (less than 8 bytes) it shall set the "Complete" bit in the reply code
  - If the matching Value requires more than one frame (greater than 7 bytes) it shall set the "Partial" bit in the reply code with the first 7 characters of the match. If the requester want the entire string it will call the node with an Addressed Message of the same type to receive as many frames as it takes for the entire *Value* string

Add New Search Criteria as String/Integer Addressed

A node should have the capability to receive and store new key/value pairs to expand and customize the nodes search status on the OLCB bus. (How do we define what is the minimum requirement for this is? [TBD] characters?). This message is defined as addressed only as adding the same key/value pair to all nodes would be useless for a search criteria. Multiple frame may be needed to transmit the entire string. The string is null terminated for both the key and the value. Upon reception the destination node shall reply with a NSP Reply message indicating failure or success in the Reply Code field with SUCCESS_COMPLETE_RESULT or one of the FAIL_ codes indicating the failure reason.

Delete Search Criteria Addressed

A node should have the capability to delete any key/value pair. This message is defined as addressed only as deleting key/value pairs in all nodes could be dangerous if used improperly. Multiple frame may be needed to transmit the entire string. The string is null terminated.

Upon reception the destination node shall reply with a NSP Reply message indicating failure or success in the Reply Code field with SUCCESS_COMPLETE_RESULT or one of the FAIL_ codes indicating the failure reason.

Return Key List Addressed

A node should have the capability to return all keys defined in a node. The maximum number of keys the destination nodes should return is passed as a parameter along with the maximum number of characters to return. This allows the sending node to manage it internal buffers efficiently. A third parameter, Offset, allows continuation of the list to be accessed from the first key. This is a 0 indexed value, i.e. the first key is Offset 0.

Upon reception the destination node shall reply with a NSP Reply message indicating failure or success in the Reply Code field with SUCCESS_COMPLETE_RESULT if all keys have been returned or SUCCESS_PARTIAL_RESULT if there are more keys available. If the call can not be completed one of the FAIL_ codes indicating the failure reason is sent.

Search By Key String/Integer Addressed

A node should have the capability to search the value of the passed key using a search mask/integer and search options to clarify how the mask should be used. The mask is purposefully not defined to allow manufacturers to implement their own search keywords, wildcards, etc. A node is not required to implement common special wildcard type characters such as "*" or "?" but doing so will allow greater user flexibility in searches.

For Integer searches the Range parameter allows searching of a number from the passed Integer up to the passed Integer + Range. For example passing 1200 with a range of 500 could result in successful matching of 1200, 1502, 1699 and 1700.

Upon reception the destination node shall reply with a NSP Reply message indicating failure or success in the Reply Code field with SUCCESS_COMPLETE_RESULT if the entire value string was returned or SUCCESS_PARTIAL_RESULT_BUFFER_TO_SHORT if the passed buffer was not long enough to hold the complete result. If the call can not be completed one of the FAIL_ codes indicating the failure reason is sent. If a match is not found the node shall not respond to the caller.

Search By Key String/Integer Global

A node should have the capability to search the value of the passed key using a search mask/integer and search options to clarify how the mask should be used. The mask is purposefully not defined to allow manufacturers to implement their own search keywords, wildcards, etc. A node is not required to implement common special wildcard type characters such as "*" or "?" but doing so will allow greater user flexibility in searches.

For Integer searches the Range parameter allows searching of a number from the passed Integer up to the passed Integer + Range. For example passing 1200 with a range of 500 could result in successful matching of 1200, 1502, 1699 and 1700.

Upon reception of a global search a node shall only send a single frame to the caller. The node shall reply with a NSP Reply message indicating failure or success in the Reply Code field with SUCCESS_COMPLETE_RESULT if the entire value string was returned or SUCCESS_PARTIAL_RESULT if the result can not fit in a single frame Result. The caller will be required to call the node with an addressed search to receive the entire string. This allows the calling node to not be overwhelmed by large blocks of data and can display partial names of search results quickly and fill in the entire string as required. If the call can not be completed one of the FAIL_ codes indicating the failure reason is sent. If a match is not found the node shall not respond to the caller.

Terminate Search

A global search that matches a large number of node can cause unacceptable delay to the user. A node that receives a Terminate Search message shall end its search request and remove any pending results from queues if possible. The Terminate Search message will have a higher CAN priority than the results so it will placed on the wire before Result messages. Depending on the hardware implementation of nodes it is highly unlikely the hardware buffer for CAN messages can be deeper than 15-16 messages and in most cases is much lower. Assuming a 500us min transfer time that mean all Result messages can be flushed within 8 ms. So making a requirement a node not transmit a Result within 50ms of receiving a Terminate Search seems like a reasonable requirement.

Upon reception the destination node shall not send a search result message after 50ms of receiving the Terminate Search message.