Control Protocols Working Group

E1.37-8, IPv4 and IPv6 Network Configuration Messages for E1.20 RDM
First Public Review comment resolutions

Reference document: E1.37-8, *IPv4 and IPv6 Network Configuration Messages for E1.20 RDM* (Document number CP/2024-1027r0)

ANSI Public review period: 02 August 2024 through 15 September 2024

Question: In your opinion, do you think the requirements of E1.37-8*, IPv4 and IPv6 Network Configuration Messages for E1.20 RDM* (DCN CP/2024-1027r0) are reasonable, and adequately address the intended subject matter?

Please answer the question using one of the options below. Select "Yes", "Yes, but…" (provide comments to support your opinion), or "No, with reasons" (the document's requirements are unacceptable or unreasonable). Note that you are offering your opinion, which is in no way construed as a vote of acceptance or approval.

Responses:

Chris Kennedy – Chamsys/Chauvet (CK)	No, with reasons
Peter Newman – Open Lighting Project (PN)	No, with reasons
Thierry Dupont – Robert Juliat (TD)	No, with reasons

Individual Comments:

No.	Commenter	Ref. section	Comment	Resolution
1	TD	Open discussion	Internally we had long debates whether we should say yes but or no. Implementing a standard instead of another one has to have solid argument on the commercial side as it will inevitably take resources, time and financial effort over other functionalities. There are very few implementations (probably a handful) of E1.37-2 among responders and even less in controllers. None of the key players in lighting desk have done it. For us we did not have the choice to use E1.37-2, it was the only mean to set IPV4 as we did not have a screen on our fixture. We have successfully implemented it 4 years ago without any particular issue among our customers. E1.37-2 has issues that you would only see during implementation. 4 years ago, only one controller had the implementation but had to be fixed before being useable with our responder. It is very hard for a controller manufacturer	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			to test an implementation without a responder and vice versa. We found in E1.37-2 3 areas that cause problems: - Apply configuration - DHCP especially in E1.33 (LLRP) - ZEROCONF LLRP (E1.33) (only 3 manufacturers worldwide have a working implementation) uses E1.37-2. It is not that easy especially on the controller side with the flaws. Zeroconf is still a problem. It means that LLRP needs to be amended when E1.37-8 is published as it supersedes E1.37-2. In LLRP should we keep the support of E1.37-2? The questions for Robert Juliat are: - E1.37-8 for a basic user point of view has the same possibilities that E1.37-See attached.2 with more features. - Do the manufacturers that implemented E1.37-2 will move to E1.37-8? - Is it going to be adopted by a larger number of manufacturers? - Maybe we should have a survey to get a sense for manufacturers would answer to the 2 questions above. - Not certain that the implementation is easier than E1.37-2 - Why not instead improving E1.37-2, correcting what is not working and adding more functionalities? - For us as we are a responder manufacturer to implement E1.37-8 we need to have a commitment for at least one controller manufacturer (setting tool) and another doing LLRP.	
2	PN	General	The draft document shows as password protected in some PDF viewers, something I've only experienced once in the past with a document draft or otherwise (E1.31 May 2024). *[TSM note: all public review documents are password protected against editing]*	See attached.
3	PN	General	In future please add draft PID values to allow trial implementations and interoperability testing of them. This has been done in the past and successfully flagged up issues which don't really become apparent until you try and implement it (including with E1.37-2 the precursor to this standard). I've done some test implementation of this standard myself but had to arbitrarily assign some test PIDs because of no formal trial ones, so wouldn't have been able to try out my work against what others had done.	See attached.
4	PN	General	Whilst IFC_INTERFACE_ENABLE presumably gets the "admin" state of an	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			interface (i.e. whether I've enabled/disabled the NIC on my laptop), there doesn't seem to be any ability to query the "operational" state of it (i.e. is a wired link up or a wireless connection present), which seems almost as important.	
5	PN	General	There either doesn't seem to be a means of always getting the current value, or there's no means of confirming that your SET has worked as intended. There will be a window where another controller may have SET new config, but hasn't COMMIT it in some way, where the system will be in an interim state. There doesn't appear to be any clarity in the document what a GET should return, the future state, or the current and someone will always want the other one so ideally it should always be possible to GET both.	See attached.
6	PN	General	I see this snuck into E1.37-7 too, which is maybe what's driving this, but most of the SET commands reply back with the Interface ID that was configured, which seems a bit redundant, given in E1.20 there's only a single controller and it ought to know what it set, and in E1.33, devices get both halves of the message anyway to put back together as required. The SET response on its own still doesn't include what was changed, so you'd still need to do a GET if you needed to make use of the data for some reason.	See attached.
7	PN	General	It would be benificial to add a PID to allow an interface/port to be identified using the interface IDs already defined in this standard (e.g. to ensure you're plugging into the expected port on a switch).	See attached.
8	PN	General	If additional setting PIDs were to be added, it feels like MTU is the next obvious missing one as it becomes more relevant with large media transfers and also the ability to upset network comms if set incorrectly.	See attached.
9	PN	General	What does IFC stand for? I'm just curious really, but its used but seemingly never defined anywhere which might help people's general understanding.	See attached.
10	TD	General	Zeroconf Topic: Usage of DHCP among end users especially small venue/concert is not popular and often having a bad reputation as it is not mastered.	See attached.
11	PN	Multiple	The following phrasing is used in many PIDs "If the interface is disabled or not present, the Responder shall NACK with a NACK Reason Code of NR_INTERFACE_UNAVAILABLE." it would be good to tie this back to the interface state which I assume is the intention? i.e. I assume i should NACK NR_INTERFACE_UNAVAILABLE if my wireless is disabled or my SFP isn't present? Presumably, and it would be good to add this, I should still NACK NR_DATA_OUT_OF_RANGE if you're trying to do something to interface 20 on a ten port switch say (i.e. an interface that's not listed in IFC_INTERFACE_ID_LIST).	See attached.
12	PN	1.3	ANSI E1.37-2 Deprecation - Type: Clarification "ANSI E1.37-2 should not be considered for new designs. Controllers may	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			wish to continue to support ANSI E1.37-2 to support legacy device implementations." Isn't there a small catch-22 here that responders may also wish to support E1.37-2 for legacy controllers?	
13	PN	E1.33 5.5	Allowed Parameter Messages - Type: Clarification Will E1.33 be updated so E1.37-8 PIDs are allowed as part of LLRP support? If not that feels like an unnecessary complication of having to keep an E1.37-2 implementation just for LLRP usage, although conversely because E1.37-8 is more fully-featured its also IMHO more complex/demanding for a basic recovery option.	See attached.
14	TD	2	A reference should be given under chapter 2 Normative References. [IPv4LL] RFC 3927– Dynamic Configuration of IPv4 Link-Local Addresses. 2005. https://datatracker.ietf.org/doc/html/rfc3927	See attached.
15	TD	2	A reference should be given under chapter 2 Normative References. [BOOTP] RFC 951 - BOOTSTRAP PROTOCOL (BOOTP) https://datatracker.ietf.org/doc/html/rfc951	See attached.
16	TD	5	Session (page8) It is difficult to say as we are not a controller manufacturer, but we are not certain that using a session is a good solution. It seems overcomplicated and might stop implementors to adopt E1.37-8. Advice from controller manufacturers would be very helpful.	See attached.
17	TD	5	Session Messages (page 8) Atomically: It is a valid word, but it is far from certain that implementor using English as a second language would understand the true meaning.	See attached.
18	CK	5	The first sentence says "To ensure continued communication between a Controller and a Responder, a session is used to make and commit changes. I don't see why a session ensures continued communication. Continued communication is determined by one or more physical cables being in place, the controller being powered on, the responder being powered on. A session does not ensure continued communication. An alternative phrasing might be "To manage" instead of "To ensure" - but better would be to more fully explain why we have chosen to use session management.	See attached.
19	CK	5	Reading this section from top to bottom it is unclear why there is a separate commit and confirm phase. Many Engineers will not be familiar with two phase commit versus three phase commit. There is no attempt to explain why three phase commit is neccessary.	See attached.
20	CK	5	It is unclear why a session is particularly important for setting IP parameters versus setting of other RDM parameters - e.g. why is it important to set both a IP address and a Subnet mask using a session whilst setting both a DMX start address and a personality does not require it. In either case a disconnect of the cable may cause only one paramter to be set and the result will be the	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			fixture will not work as expected. If other parameters would benefit from this then we should be designing a generic session management for RDM.	
21	CK	5	If we keep the separate commit and confirm then the sentence "a session is used to make and commit changes" should be expanded to "a session is used to make, commit and confirm changes"	See attached.
22	CK	5	I am concerned that the session management is burdensome to Responders and will discourage implementation. In standard RDM, Responders manufacturers do not need to maintain state information beyond the configuration information they already store within their fixtures for non RDM operation. Implementing RDM requires simple handlers that process the message, action it using their pre-existing configuration functions that they use on their displays and then send a response. The session management adds a whole new level of complexity to Responders where they need to store all the requests and to implement rollback routines. Whilst fixtures that support Ethernet are likely to be modern micro controllers with reasonable amounts of memory, this will the not always be the case. The majority of fixtures only have a single network port so this is overkill. Session management should be recommended but optional.	See attached.
23	CK	5	COMMIT_RUNNING and COMMIT_RUNNING_PERSISTENT. Responders may always commit changes to their persistent memory - either due to choice or neccesity. I see no way in the document for the Responder to indicate whether it supports separate store to running and persistent memory. By forcing support for COMMIT_RUNNING_PERSISTENT you are forcing Responders to store all their network data twice in Persistent storage since if the Responder is powered off before the Confirm you must be able to roll back. This massively changes their existing persistent storage architecture. My preferred option is to simplify by consolidating to a single COMMIT option and make it a Responder choice how a power off prior to Confirm is handled. This could be declared in a capability in the GET_SESSION_CONTROL response.	See attached.
24	PN	5.1	Get/Set Session Control (IFC_SESSION_CONTROL) "Make each change required, including the received Session ID with each message." Where/how is the session ID included in each message? Please include a full example with some actual bytes to clarify its usage. I can't see it in the PD info for general PIDs like IFC_IPV4_ADDRESS so is it elsewhere? Likewise Figure 5-1 says "SET Message with Session ID".	See attached.
25	PN	5.1	Get/Set Session Control (IFC_SESSION_CONTROL) Put session operations after time or before session ID so set request structure is closer to get response to allow people possible efficiencies when processing.	See attached.

No.	Commenter	Ref. section	Comment	Resolution
26	PN	5.1	Get/Set Session Control (IFC_SESSION_CONTROL) Says "Session Operations is a bit field indicating whether to commit or discard any pending changes or confirm previously applied changes." but then the definition doesn't specifiy it as bits, it's just an enum. e.g. the LSB isn't discard/commit etc. Also doesn't include the usual line about what controller/responder should do with unspecified bits.	See attached.
27	CK	6.1	Get Interface List (IFC_INTERFACE_ID_LIST) The interface identifiers returned are known as Interface IDs and shall be used by other parameters to uniquely identify an interface until an IFC_INTERFACE_TOPOLOGY queued message is received.	See attached.
28	CK	6.1	Please indicate whether heirarchy changes can re-use the same Interface IDs?	See attached.
29	CK	6.1	Assuming IDs can bre re-used, what happens if the heirarchy has changed but the Controller sends a parameter message before receiving the queued message? Potentially the Controller will receive data for the wrong Interface ID? Should all messages using Interface IDs be NACKed after a hierarchy change until the Controller requests a new Get Interface List?	See attached.
30	PN	6.1	Explicitly clarify that each interface ID should only be listed once (I assume that's the case) e.g. if you're modelling a switch with multiple ports (interfaces) in the same VLAN, presumably each port should be listed, and the VLANs?	See attached.
31	PN	6.1	What happens if we've got an interface 2 and a VLAN 2? The standard doesn't seem to explicitly mention it and RFC 3493 doesn't mention VLANs. In fact "4.3 Sample Interface Hierarchy[Informative]" even includes physical interface 1 and VLAN 1.	See attached.
32	PN	6.1	It would probably be good to have a general section on interface IDs specifying that 0 is reserved and also dealing with the max ID, plus the VLAN stuff mentioned above). E1.37-2 had "The interface identifiers returned shall range from 1 to 0xFFFFFF00." so it may be beneficial to track that upper limit across for compatibility". Sadly the E1.37-2 reserved block at the top isn't big enough for all the VLANs, but having a block/mask at the top for the VLANs may be a good solution (so VLAN ID | 0xFFFFF000).	See attached.
33	PN	6.1	Add a line that "Interface identifiers are not required to be contiguous.", this both matches the old E1.37-2 spec and allows implementers to just take their native IDs and use them without having to remap them (and risk introducing errors).	See attached.
34	PN	6.2	Get Response Interface Topology (IFC_INTERFACE_TOPOLOGY) The standard says "This parameter is intended for use with queued messages. Controllers shall not issue a GET_COMMAND for this parameter.	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			Responders receiving a GET_COMMAND for this parameter shall NACK with a NACK Reason Code of NR_GET_QUEUED_ONLY." but then later on "Responders receiving a GET_COMMAND shall respond with a GET_COMMAND_RESPONSE with the value of Topology Changed (Bit 0) set to 0.". Which is it? I'm not aware of any other PIDs we can't GET, but can expect a response from, but I might have forgotten one.	
35	CK	6.2	Get Response Interface Topology (IFC_INTERFACE_TOPOLOGY) This queued message response is used for two distinct cases - topology has changed and on initial boot. There is no way to distinguish between the two cases. On boot it is quite normal that the topology will not have changed. A controller that remains permanently on will therefore get notifications of topology change every time the lights are powered on and off.	See attached.
36	CK	6.3	Get Interface Parent (IFC_INTERFACE_PARENT) Paragraph "Parent ID: The identifier of the parent interface of the interface, or 0 (0x00000000), indicating the interface is on the root." Please add "of the Responder's interface tree" to distinguish from Root sub ids.	See attached.
37	PN	6.4	Get/Set Interface Enable(IFC_INTERFACE_ENABLE) "Some Responders may choose to include Interface IDs for interfaces which are not present in the list of Interface IDs returned from GET:IFC_INTERFACE_ID_LIST. For example: ports which receive SFP (small form-factor pluggable) modules on a network switch. In such cases, a Responder shall return NOT_PRESENT in GET_COMMAND_RESPONSE." Maybe be more explicit that it's SFPs that aren't fitted. Also clarify what should happen (presumably a NACK) if I try and GET an interface that isn't in the IFC_INTERFACE_ID_LIST? What should happen if I SET a not present interface, the same NACK?	See attached.
38	PN	6.4	Add a line that responders may wish to queue an updated state after a bounce when the interface is re-enabled.	See attached.
39	CK	6.4	Get/Set Interface Enable (IFC_INTERFACE_ENABLE) Clarify that rolling back a Bounce does not require the Responder to re-perform the Bounce?	See attached.
40	CK	6.4	Should Time be less than the Commit Time?	See attached.
41	CK	6.4	The sentence "Some Responders may choose to include Interface IDs for interfaces which are not present in the list of Interface IDs returned from GET:IFC_INTERFACE_ID_LIST." can be read to suggest that Responders can reply to IDs that are not in the GET_IFC_INTERFACE_ID_LIST". Reorder as "Some Responders may choose to include Interface IDs in the list of Interface IDs returned from GET:IFC_INTERFACE_ID_LIST for interfaces which are not present."	See attached.
42	PN	6.5	Get Interface Hardware Address (IFC_INTERFACE_HARDWARE_ADDRESS)	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			Mirror the ARP protocol's hardware addresses as the single biggest/most future-proofed list of hardware addresses. i.e. make the hardware address type field 16 bit and use EUI-48 as type 1 and EUI-64 as type 27. This leaves the standard future-proofed to add additional hardware types by just including their lengths in the table and reduces the risk of programmers mistakes mapping from a widely used standard to the otherwise arbitrary assignments used in this standard. As well as the risk of confusion between E1.37-2 which used ARP's hardware types (where EUI-48/MAC is 1) and this standard (where it's zero).	
43	PN	6.5	Given (from what I can tell) at least some EUI-64 addresses are derived from EUI-48 addresses, its presumably possible for a device to have both, so shouldn't we be able to specify which one we wish to return during the GET request?	See attached.
44	PN	6.5	By having combined the interface hardware address fetching into one PID (compared to E1.37-2's explicit INTERFACE_HARDWARE_ADDRESS_TYPE1 PID) then I don't believe E1.37-5's JSON schema, or at least some of the other alternative PID schemas, have the ability to have different interpretation formats for the same range of bytes depending on either their length or an associated field, which means controllers have to write more special purpose code, or users get a worse experience if e.g. their MAC address doesn't turn up nicely fomatted.	See attached.
45	CK	6.6	Get Interface Fixed Label (IFC_INTERFACE_FIXED_LABEL Fixed Label. References section 3.3 which then references RDM 10.1 which says that unless otherwise specified the length is 32 bytes. Change to "The Fixed Label is a text field as defined in Section 3.3 variable up to 227 bytes".	See attached.
46	CK	6.6	Would it be better to call it "IFC_INTERFACE_MANUFACTURER_LABEL" rather than "IFC_INTERFACE_FIXED_LABEL" to avoid the confusion between Fixed Label and Fixed Length?	See attached.
47	CK	6.7	Get/Set Interface User Assignable Label (IFC_INTERFACE_USER_LABEL) Remove "This parameter is intended for user-definable labels." as it repeats the first paragraph.	See attached.
48	CK	6.7	User Label. References section 3.3 which then references RDM 10.1 which says that unless otherwise specified the length is 32 bytes. Change to "The User Label is a text field as defined in Section 3.3 variable up to 227 bytes".	See attached.
49	PN	6.7	Get/Set Interface User Assignable Label(IFC_INTERFACE_USER_LABEL) What if anything should the IFC_INTERFACE_USER_LABEL default to? Blank is potentially unhelpful (as controllers need to do some magic) but matching IFC_INTERFACE_FIXED_LABEL is potentially confusing.	See attached.
50	PN	6.7	Why does SET IFC_INTERFACE_USER_LABEL return the interface ID?	See attached.
51	PN	6.11	Get Interface Duplex Mode(IFC_INTERFACE_DUPLEX_MODE)	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			Doesn't allow setting duplex mode.	
52	PN	6.11	I assume this is intended to return the capability of an interface regarding duplex, rather than the state its currently in? You really need both to make a good judgement as to whether it's operating optimally in case something has auto-negotiated to simplex.	See attached.
53	CK	6.12	Get/Set Energy Efficient Ethernet Enable (IFC_INTERFACE_EEE) How do we know if the interface supports EEE? Should there be a capability? Noting that some ports may support it whilst others do not.	See attached.
54	CK	6.12	If interface does not support EEE and Controller attempts to set it then should it be NACKed?	See attached.
55	CK	6.13	Get/Set Interface Power Enable (IFC_INTERFACE_POWER_ENABLE) Reference IFC_INTERFACE_POWER_STATUS to determine whether port supports power	See attached.
56	CK	6.13	Could we move IFC_INTERFACE_POWER_STATUS before IFC_INTERFACE_EEE so that we don't have to go searching for it when reading document top to bottom?	See attached.
57	PN	6.13	Get/Set Interface Power Enable (IFC_INTERFACE_POWER_ENABLE) It would be good to add the bounce options from IFC_INTERFACE_ENABLE to enable a PoE fed device to be easily bounced, which potentially could be in the control path to the controller.	See attached.
58	PN	6.14	"The reserved value of 0xFFFF shall be reported in the Power Requested, Power Granted, or Power Consumedif the interface does not have the required information available to report." This value isn't defined as a constant in the defined parameters like they normally are.	See attached.
59	PN	6.14	"The reserved value of 0xFFFF shall be reported in the Power Requested, Power Granted, or Power Consumed if the interface does not have the required information available to report." Any reason "Maximum Power Capability" isn't covered by this case too? I could imagine some modular device and a third party PoE chipset where it might not be able to read that info either (or varies by hardware) but it at least knows its power sourcing.	See attached.
60	CK	6.14	Get Interface Power Status (IFC_INTERFACE_POWER_STATUS) Many systems have a total maximum power across all network ports separate to individual port limits which is the most important limit. We can calculate the total used by summing each port, but there does not seem to be a method for reporting the total maximum power.	See attached.
61	CK	7	IPv4 Configuration Messages It should be allowed to set IP network settings for interfaces that are disabled and may in some cases be desireable to do it before enabling the port. Compare with PC. Arguably also for interfaces that are not available. Could	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			we change to "If the interface is disabled or not present, the Responder may choose to NACK with a NACK Reason Code of NR_INTERFACE_UNAVAILABLE."	
62	PN	7.1	Get/Set IPv4 Address (IFC_IPV4_ADDRESS) No mention of what to do in ACK_OVERFLOW; presumably drop the interface ID on subsequent packets? Or is it not supported?	See attached.
63	PN	7.1	If ACK_OVERFLOW isn't supported, SET PDL should be reduced as we can currently SET more than we can read back via GET (due to the 1 byte overhead of source when GETing).	See attached.
64	PN	7.1	If an IP has been assigned via DHCP (for example) does SETting one or more others just set them statically and remove the DHCP assignment? This isn't clarified anywhere. Likewise if I wish to add another "alias"/"virtual" IP I assume I need to SET both the existing and new one to make them both stay?	See attached.
65	PN	7.2	Windows, somewhat bafflingly, allows me to set multiple gateways on one interface; which should this return, the first?	See attached.
66	PN	7.2	IFC_IPV4_GATEWAY allows me to GET or SET a gateway per interface, but doesn't give me any indication of the routing table on the device and which will win and be the default route.	See attached.
67	PN	7.2	Get/Set IPv4 Gateway (IFC_IPV4_GATEWAY) What value should be returned for IPv4 Gateway if it's not set/has been removed with IFC_IPV4_GATEWAY_REMOVE?	See attached.
68	PN	7.3	Set Remove IPv4 Gateway (IFC_IPV4_GATEWAY_REMOVE) Given IFC_IPV4_GATEWAY has to have a way to represent a non-existent gateway in 4 bytes, why not make this PID redundant and just set the magic value to IFC_IPV4_GATEWAY to remove any configured gateway? This would also make that PID more symmetric and remove this non-standard one.	See attached.
69	CK	7.3	Set Remove IPv4 Gateway (IFC_IPV4_GATEWAY_REMOVE) It is unclear why this PID is required rather than using IFC_IPV4_GATEWAY with zero length PDL. For IP addresses there is IFC_IPV4_ADDRESS and no equivalent remove PID. The same comment applies to IFC_IPV6_GATEWAY. Either remove the PID or explain.	See attached.
70	PN	7.4	Get/Set IPv4 DNS Address (IFC_IPV4_DNS) On a lot of Linux systems (for example) DNS servers can either be configured at the interface level or centrally, although are processed from the central list, so the interface a server came from may not be easily known (and might not necessarily be directly within its subnet). What should be returned in this situation?	See attached.
71	PN	7.4	"If more IPv4 DNS server addresses are provided than can be accepted, the Responder shall NACK with a NACK Reason Code of NR_IPV4_LIMIT_EXCEEDED." What should happen in the unlikely event the	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			responder has more configured than can be returned (56)?	
72	CK	7.6	Get/Set IPv4 Mode (IFC_IPV4_MODE) The capabilities retrievable with IFC_IPV4_SUPPORTED_MODES are not the same as the modes assignable with IFC_IPV4_MODE. It is likely that many fixtures will support both STATIC and DHCP but will not support fallback. Therefore despite getting IFC_IPV4_SUPPORTED_MODES the set of IFC_IPV4_MODE will fail with NR_MODE_UNSUPPORTED. IFC_IPV4_SUPPORTED_MODES should be modified to provide the full capabilities. Also applies to IFC_IPV6_SUPPORTED_MODES.	See attached.
73	PN	7.7	Get DHCPv4 Server Result (IFC_IPV4_DHCP_RESULT) Change "DHCPv4 Address" to "DHCPv4 Server Address" to avoid confusion with people putting the DHCP assigned address, rather than the server's address in the field.	See attached.
74	PN	8.5	Get/Set IPv6 DNS Address (IFC_IPV6_DNS) PDL or GET Response and SET Request should be "Variable (0x04-0xE4) 0x04 + (Multiple of 0x10)", rather than 0xDC and 0x12.	See attached.
75	PN	8.8	Get DHCPv6 Server Result (IFC_IPV6_DHCP_RESULT) Change "DHCPv6 Address" to "DHCPv6 Server Address" to avoid confusion with people putting the DHCP assigned address, rather than the server's address in the field.	See attached.
76	PN	9.1	Get DNS Capability(IFC_DNS_CAPABILITIES) ""Note: Depending on the Responder’s operating system, DNS configuration may only exist on the root device (0x0000), while in other cases it may be per-interface."" I'm unclear why this note is just attached to DNS capability? Surely depending on how the device is ""modelled"" it may only have network interfaces on the root device or on sub-devices. Or is this actually meaning something about particular interfaces rather than sub-devices? It certainly seems more likely for most devices that they will have a single DNS hostname (etc) rather than one per interface. Is this note actually supposed to mean you should return and interact with the reserved root interface (0x00000000) from the hierarchy tree to set the hostname globally? If so that needs a lot more flagging if the root of the tree has a set of properties here but doesn't appear in the main list of interfaces elsewhere?	See attached.
77	CK	9.1	Get DNS Capability (IFC_DNS_CAPABILITIES) The comment "Note: Depending on the Responder’s operating system, DNS configuration may only exist on the root device (0x0000), while in other cases it may be per-interface." Does Root refer to the Root sub device or the root of the Interface tree? The fact that it has a 4 digit 0x0000 sugests Sub Device not Interface ID.	See attached.
78	PN	9.2	Get/Set DNS Hostname(Label)(IFC_DNS_LABEL)	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			PDL or GET Response and SET Request should be "Variable (0x05 –0x43)", rather than 0x04 as the DNS label has to be a minimum of one character.	
79	CK	10.3	Get/Set VLAN Label (IFC_VLAN_LABEL) Label. References section 3.3 which then references RDM 10.1 which says that unless otherwise specified the length is 32 bytes. Change to "The Label is a text field as defined in Section 3.3 variable up to 227 bytes".	See attached.
80	PN	10.3	Doesn't this PID clash with IFC_INTERFACE_ID_LIST as that wanted VLANs in the main list, as 32 bit interface IDs, but this has them as 16 bit padded entries of a particular interface ID?	See attached.
81	PN	10.3	"For SET_COMMAND, if the Label string is longer than supported, the Responder shall NACK with a NACK Reason Code of NR_DATA_OUT_OF_RANGE as defined in [RDM]." Pretty certain that particular edge case would be a format error in RDM, whereas trying to set a VLAN label for VLAN 5000 would be data out of range.	See attached.
82	CK	Appendix A : Defined Parameters	None of the PIDs are marked as required, but section 6 implies that IFC_SESSION_CONTROL is required if other SET_PARAMETERS messages are supported.	See attached.
83	CK	Appendix A : Defined Parameters	IFC_INTERFACE_LIST and IFC_INTERFACE_TOPOLOGY are required if other network parameter messages are supported.	See attached.
84	CK	Appendix A : Defined Parameters	The standard provides no generic way for identifying the function of a network port with respect to the fixture. Nearly all fixtures will have a network port that supports the primary control of the fixture using Art-Net/sACN - but there is no generic way from these PIDs of detecting which port in the hierarchy it is. There should be a numeric Interface Function type defined in the standard with e.g. Primary control port, Secondary control port.	See attached.
85	PN	Appendix A: Defined Parameters	Why is packing universally disallowed? This will make for a very slow way to query a switch type device with lots of interfaces...	See attached.
86	PN	Table A-2: Defines	I believe the current SESSION_WINDOW duration is too short for manual use, e.g. someone using a simpler controller where they might be having to manually enter bytes (or even something command line based or similar that does some amount of pretty-printing), anything without a custom UI which handles the session ID management automagically. From what I currently understand (but as mentioned elsewhere some of the session based stuff is very unclear), you'd need to do at least the following (for example, just to make the most basic change without sawing off a branch): GET IFC_SESSION_CONTROL SET IFC_IPV4_ADDRESS	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			SET IFC_IPV4_GATEWAY SET IFC_SESSION_CONTROL COMMIT_RUNNING	
87	PN	Table A-5: Interface States	The table only lists a description for Set for "BOUNCE", whereas the words for 6.4 say "For GET_COMMAND_RESPONSE, returns whether the interface is enabled, disabled or not present. BOUNCE shall not be returned, unless a previous SET_COMMAND using BOUNCE is still in progress.". Likewise unknown lists a get description, but isn't listed in 6.4.	See attached.
88	PN	Table A-7: Interface Types	Does SFP fall under "COMBINATION", maybe add it as an explicit example? What about those switches where some ports are shared between say an SFP cage and an Ethernet port? Should one of those be a separate category?	See attached.
89	PN	Table A-10: Power States	First column header is "Interface State" rather than say "Power State"	See attached.
90	PN	Table A-11: IFC Power Type	First column header is "Interface State" rather than say "IFC Power Type"	See attached.
91	PN	Table A-12: IFC Power Role	First column header is "Interface State" rather than say "IFC Power Role"	See attached.
92	TD	Table A-13	Zeroconf should be explained in the context of E1.37-8. Not all implementors are expert in networking. We should provide examples showing the process that would avoid interpretation. For the less experienced implementor explaining the relationship between DHCP and Zeroconf or BOOTP would be very helpful.	See attached.
93	TD	Table A-13	BOOTP should be explained with examples/scenarios as it is not very known.	See attached.
94	PN	Table A-13: IPv4 Sources	A value should be added to represent DHCP unknown. As I said during the writing of E1.37-2, "It’s very hard on Linux to find out if the current address was assigned by DHCP or not (at least 4 userspace programs on Debian); we can’t partially support the PID, but want to expose the core IP data, so what about either splitting DHCP status off to a separate PID, or adding an extra state to DHCP of unknown, for when the software can’t tell."	See attached.
95	TD	Table A-14: IPv4 Supported Modes	It says: “The interface supports assigning an IPv4 address using some zero-configuration method” The word “some” for an implementor is ambiguous and should be clarified. Are we talking about method listed in Table A-15: IPv4 Modes? We guess it is but we can be wrong.	See attached.
96	TD	Table A-15-Modes	We would suggest giving examples/scenarios to explain what the modes are. It might be very clear for a network expert, but it might not be the case for implementors.	See attached.
97	PN	Table A-15: IPv4 Modes	""A static IPv4 address assignment shall be used, with no fallback."" and ""An attempt to assign an IPv4 address using zero-configuration shall be performed	See attached.

No.	Commenter	Ref. section	Comment	Resolution
			first, with no fallback."" surely both static and (probably) Zeroconf can't fail (well assuming there's a static address configured in the first case), so the concept of their being a fallback doesn't really make sense? Calling them (and DHCP_NONE and BOOTP_NONE) <name>_ONLY might make this clearer?	
98	PN	Table A-19: IPv6 Mode Defines	First column header is "Method Define" rather than say "IPv6 Mode"	See attached.
99	PN	Table A-22: VLAN Tagged VLAN Flags	First column header currently mirrors the previous table and should be something more specific, e.g. "VLAN Tagged VLAN"	See attached.
100	PN	Table A-23: Additional Response NACK Reason Codes	Whilst admittedly we aren't particularly numerically challenged in terms of NACK codes with 16 bits available and a tiny fraction used, NR_IPV4_LIMIT_EXCEEDED/NR_IPV6_LIMIT_EXCEEDED/NR_VLAN_LIMIT_EXCEEDED do feel rather specific and a bit of a burden for controllers to implement, whilst also not being very flexible for other future uses or manufacturer specific re-usage. Something more generic like NR_TOO_MANY_ENTRIES (or some better wording/naming) would achieve the same thing in a more generic way. It's also not like IPv4 and IPv6 addresses appear in the same PID for example, which would make sense of why these need disambiguating.	See attached.
101	PN	Table A-23: Additional Response NACK Reason Codes	Similar comment to above, but isn't NR_MODE_UNSUPPORTED just NR_DATA_OUT_OF_RANGE like you'd get if you tried to set an invalid DMX_PERSONALITY for example?	See attached.

Number:

1

Commenter:

Thierry Dupont

Section Number:

Discussion

Section Title:

-

Description:

Internally we had long debates whether we should say yes but or no. Implementing a standard instead of another one has to have solid argument on the commercial side as it will inevitably take resources, time and financial effort over other functionalities. There are very few implementations (probably a handful) of E1.37-2 among responders and even less in controllers. None of the key players in lighting desk have done it. For us we did not have the choice to use E1.37-2, it was the only mean to set IPV4 as we did not have a screen on our fixture. We have successfully implemented it 4 years ago without any particular issue among our customers. E1.37-2 has issues that you would only see during implementation. 4 years ago, only one controller had the implementation but had to be fixed before being useable with our responder. It is very hard for a controller manufacturer to test an implementation without a responder and vice versa. We found in E1.37-2 3 areas that cause problems: - Apply configuration - DHCP especially in E1.33 (LLRP) - ZEROCONF LLRP (E1.33) (only 3 manufacturers worldwide have a working implementation) uses E1.37-2. It is not that easy especially on the controller side with the flaws. Zeroconf is still a problem. It means that LLRP needs to be amended when E1.37-8 is published as it supersedes E1.37-2. In LLRP should we keep the support of E1.37-2? The questions for Robert Juliat are: - E1.37-8 for a basic user point of view has the same possibilities that E1.37-2 with more features. - Do the manufacturers that implemented E1.37-2 will move to E1.37-8? - Is it going to be adopted by a larger number of manufacturers? - Maybe we should have a survey to get a sense for manufacturers would answer to the 2 questions above. - Not certain that the implementation is easier than E1.37-2 - Why not instead improving E1.37-2, correcting what is not working and adding more functionalities? - For us as we are a responder manufacturer to implement E1.37-8 we need to have a commitment for at least one controller manufacturer (setting tool) and another doing LLRP.)

Resolution:

No Action

Resolution Description:

The data model that underlies E1.37-2 is overly restrictive as it could not descirbe multiple network interfaces within a single responder. Instead of making significant changes to E1.37-2 which would break compatibility the task group elected to create a new standard. This path was chosen to assist implementors rather than to create friction.

Number:

2

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

The draft document shows as password protected in some PDF viewers, something I've only experienced once in the past with a document draft or otherwise (E1.31 May 2024). [TSM note: all public review documents are password protected against editing])

Resolution:

No Action

Resolution Description:

All public review documents are protected against editing.

Number:

3

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

In future please add draft PID values to allow trial implementations and interoperability testing of them. This has been done in the past and successfully flagged up issues which don't really become apparent until you try and implement it (including with E1.37-2 the precursor to this standard). I've done some test implementation of this standard myself but had to arbitrarily assign some test PIDs because of no formal trial ones, so wouldn't have been able to try out my work against what others had done.)

Resolution:

Accept

Resolution Description:

Agreed. We have added draft PIDs to the document for the next review.

Number:

4

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

Whilst IFC_INTERFACE_ENABLE presumably gets the "admin" state of an interface (i.e. whether I've enabled/disabled the NIC on my laptop), there doesn't seem to be any ability to query the "operational" state of it (i.e. is a wired link up or a wireless connection present), which seems almost as important.)

Resolution:

Accept in Part

Resolution Description:

We have added clarifying text to IFC_INTERFACE_LINE_SPEED allowing the "operational" state to be queried.

Number:

5

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

There either doesn't seem to be a means of always getting the current value, or there's no means of confirming that your SET has worked as intended. There will be a window where another controller may have SET new config, but hasn't COMMIT it in some way, where the system will be in an interim state. There doesn't appear to be any clarity in the document what a GET should return, the future state, or the current and someone will always want the other one so ideally it should always be possible to GET both.)

Resolution:

Reject

Resolution Description:

There is no method for getting the values set as part of a Session Change Set. However, we have added clarifying text to each parameter that can be set with a GET_COMMAND_RESPONSE indicating the value that will be returned.

Number:

6

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

I see this snuck into E1.37-7 too, which is maybe what's driving this, but most of the SET commands reply back with the Interface ID that was configured, which seems a bit redundant, given in E1.20 there's only a single controller and it ought to know what it set, and in E1.33, devices get both halves of the message anyway to put back together as required. The SET response on its own still doesn't include what was changed, so you'd still need to do a GET if you needed to make use of the data for some reason.)

Resolution:

Reject

Resolution Description:

The Interface ID is a sub-address mechanism, and this was an intentional addition to retain consistency with other standards that have a similar mechanism.

Number:

7

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

It would be benificial to add a PID to allow an interface/port to be identified using the interface IDs already defined in this standard (e.g. to ensure you're plugging into the expected port on a switch).)

Resolution:

Accept

Resolution Description:

The task group agrees this will be useful and it has been added to the document.

Number:

8

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

If additional setting PIDs were to be added, it feels like MTU is the next obvious missing one as it becomes more relevant with large media transfers and also the ability to upset network comms if set incorrectly.)

Resolution:

Reject

Resolution Description:

The task group believes this is out of scope for this standard and there are other more appropriate solutions.

Number:

9

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

What does IFC stand for? I'm just curious really, but its used but seemingly never defined anywhere which might help people's general understanding.)

Resolution:

Reject

Resolution Description:

Interface Configuration. We wanted something that was not already overloaded such as IF and that helps group this family of PIDs together. The task group does not believe this needs to be defined because it is simply a define string.

Number:

10

Commenter:

Peter Newman

Section Number:

General

Section Title:

-

Description:

Zeroconf Topic: Usage of DHCP among end users especially small venue/concert is not popular and often having a bad reputation as it is not mastered.)

Resolution:

Reject

Resolution Description:

No action requested. There are applications where Static, ZeroConf, and DHCP are each applicable.

Number:

11

Commenter:

Peter Newman

Section Number:

Multiple

Section Title:

-

Description:

The following phrasing is used in many PIDs "If the interface is disabled or not present, the Responder shall NACK with a NACK Reason Code of NR_INTERFACE_UNAVAILABLE." it would

be good to tie this back to the interface state which I assume is the intention? i.e. I assume i should NACK NR_INTERFACE_UNAVAILABLE if my wireless is disabled or my SFP isn't present? Presumably, and it would be good to add this, I should still NACK NR_DATA_OUT_OF_RANGE if you're trying to do something to interface 20 on a ten port switch say (i.e. an interface that's not listed in IFC_INTERFACE_ID_LIST).)

Resolution:

Accept in Principle

Resolution Description:

We added clarification that the this should be used when the interface was provided in the list of interfaces, otherwise NR_DATA_OUT_OF_RANGE is appropriate.

Number:

12

Commenter:

Peter Newman

Section Number:

1.3

Section Title:

ANSI E1.37-2

Description:

ANSI E1.37-2 Deprecation - Type: Clarification "ANSI E1.37-2 should not be considered for new designs. Controllers may wish to continue to support ANSI E1.37-2 to support legacy device implementations." Isn't there a small catch-22 here that responders may also wish to support E1.37-2 for legacy controllers?)

Resolution:

Accept in Principle

Resolution Description:

The text has been changed to remove references to controllers and devices.

Number:

13

Commenter:

Peter Newman

Section Number:

E1.33 5.5

Section Title:

-

Description:

Allowed Parameter Messages - Type: Clarification Will E1.33 be updated so E1.37-8 PIDs are allowed as part of LLRP support? If not that feels like an unnecessary complication of having to keep an E1.37-2 implementation just for LLRP usage, although conversely because E1.37-8 is more fully-featured its also IMHO more complex/demanding for a basic recovery option.)

Resolution:

No Action

Resolution Description:

E1.33 will be updated during the next revision to reference E1.37-8 PIDs.

Number:

14

Commenter:

Thierry Dupont

Section Number:

2

Section Title:

Normative References

Description:

A reference should be given under chapter 2 Normative References. [IPv4LL] RFC 3927– Dynamic Configuration of IPv4 Link-Local Addresses. 2005. https://datatracker.ietf.org/doc/html/rfc3927)

Resolution:

Accept in Principle

Resolution Description:

This has been added to the document.

Number:

15

Commenter:

Thierry Dupont

Section Number:

2

Section Title:

Normative References

Description:

A reference should be given under chapter 2 Normative References. [BOOTP] RFC 951 - BOOTSTRAP PROTOCOL (BOOTP) https://datatracker.ietf.org/doc/html/rfc951)

Resolution:

Accept in Part

Resolution Description:

We have removed all references to BOOTP in the document.

Number:

16

Commenter:

Thierry Dupont

Section Number:

5

Section Title:

Session Messages

Description:

Session (page8) It is difficult to say as we are not a controller manufacturer, but we are not certain that using a session is a good solution. It seems overcomplicated and might stop implementors to adopt E1.37-8. Advice from controller manufacturers would be very helpful.)

Resolution:

Reject

Resolution Description:

The session provides very important protections in a multi-controller environment, as well as supporting atomic set requirements. We have discussed this at length in the task group and agreed on this mechanism. We have updated the session mechansim to make this easier for simpler Responders.

Number:

17

Commenter:

Thierry Dupont

Section Number:

5

Section Title:

Session Messages

Description:

Session Messages (page 8) Atomically: It is a valid word, but it is far from certain that implementor using English as a second language would understand the true meaning.)

Resolution:

Reject

Resolution Description:

While the task group recognizes this term is not neccessarily clear to non-native English speakers, it has a specific meaning in software engineering which means its use is relevant.

Number:

18

Commenter:

Chris Kennedy

Section Number:

5

Section Title:

Session Messages

Description:

The first sentence says "To ensure continued communication between a Controller and a Responder, a session is used to make and commit changes. I don't see why a session ensures continued communication. Continued communication is determined by one or more physical cables being in place, the controller being powered on, the responder being powered on. A session does not ensure continued communication. An alternative phrasing might be "To manage" instead of "To ensure" - but better would be to more fully explain why we have chosen to use session management.)

Resolution:

Accept in Principle

Resolution Description:

We have clarified the text in the document.

Number:

19

Commenter:

Chris Kennedy

Section Number:

5

Section Title:

Session Messages

Description:

Reading this section from top to bottom it is unclear why there is a separate commit and confirm phase. Many Engineers will not be familiar with two phase commit versus three phase commit. There is no attempt to explain why three phase commit is neccessary.)

Resolution:

Accept in Principle

Resolution Description:

We added an explanation of the commit stage, using the display/monitor configuration process found in many operating systems.

Number:

20

Commenter:

Chris Kennedy

Section Number:

5

Section Title:

Session Messages

Description:

It is unclear why a session is particularly important for setting IP parameters versus setting of other RDM parameters - e.g. why is it important to set both a IP address and a Subnet mask using a session whilst setting both a DMX start address and a personality does not require it. In either case a disconnect of the cable may cause only one paramter to be set and the result will be the fixture will not work as expected. If other parameters would benefit from this then we should be designing a generic session management for RDM.)

Resolution:

Reject

Resolution Description:

The requirement to set parameters atomically, as well as manage multi-controller sessions means sessions are an integral part of this standard.

Number:

21

Commenter:

Chris Kennedy

Section Number:

5

Section Title:

Session Messages

Description:

If we keep the separate commit and confirm then the sentence "a session is used to make and commit changes" should be expanded to "a session is used to make, commit and confirm changes")

Resolution:

Accept

Resolution Description:

Implemented in the document.

Number:

22

Commenter:

Chris Kennedy

Section Number:

5

Section Title:

Session Messages

Description:

I am concerned that the session management is burdensome to Responders and will discourage implementation. In standard RDM, Responders manufacturers do not need to maintain state information beyond the configuration information they already store within their fixtures for non RDM operation. Implementing RDM requires simple handlers that process the message, action it using their pre-existing configuration functions that they use on their displays and then send a response. The session management adds a whole new level of complexity to Responders where they need to store all the requests and to implement rollback routines. Whilst fixtures that support Ethernet are likely to be modern micro controllers with reasonable amounts of memory, this will the not always be the case. The majority of fixtures only have a single network port so this is overkill. Session management should be recommended but optional.)

Resolution:

Reject

Resolution Description:

The session provides very important protections in a multi-controller environment, as well as supporting atomic set requirements. We have discussed this at length in the task group and agreed on this mechanism. We have updated the session mechansim to make this easier for simpler Responders.

Number:

23

Commenter:

Peter Newman

Section Number:

5

Section Title:

Session Messages

Description:

COMMIT_RUNNING and COMMIT_RUNNING_PERSISTENT. Responders may always commit changes to their persistent memory - either due to choice or neccesity. I see no way in the document for the Responder to indicate whether it supports separate store to running and persistent memory. By forcing support for COMMIT_RUNNING_PERSISTENT you are forcing Responders to store all their network data twice in Persistent storage since if the Responder is powered off before the Confirm you must be able to roll back. This massively changes their existing persistent storage architecture. My preferred option is to simplify by consolidating to a single COMMIT option and make it a Responder choice how a power off prior to Confirm is handled. This could be declared in a capability in the GET_SESSION_CONTROL response.)

Resolution:

Reject

Resolution Description:

The session provides very important protections in a multi-controller environment, as well as supporting atomic set requirements. We have discussed this at length in the task group and agreed on this mechanism. We have updated the session mechansim to make this easier for simpler Responders.

Number:

24

Commenter:

Peter Newman

Section Number:

5.1

Section Title:

Get/Set Session Control

Description:

Get/Set Session Control (IFC_SESSION_CONTROL) "Make each change required, including the received Session ID with each message." Where/how is the session ID included in each message? Please include a full example with some actual bytes to clarify its usage. I can't see it in the PD info for general PIDs like IFC_IPV4_ADDRESS so is it elsewhere? Likewise Figure 5-1 says "SET Message with Session ID".)

Resolution:

Accept in Principle

Resolution Description:

The Session ID should have been provided with each set message. These have now been added.

Number:

25

Commenter:

Peter Newman

Section Number:

5.1

Section Title:

Get/Set Session Control

Description:

Get/Set Session Control (IFC_SESSION_CONTROL) Put session operations after time or before session ID so set request structure is closer to get response to allow people possible efficiencies when processing.)

Resolution:

Reject

Resolution Description:

The task group does not believe this warrants changes as it would not offer significant benefits.

Number:

26

Commenter:

Peter Newman

Section Number:

5.1

Section Title:

Get/Set Session Control

Description:

Get/Set Session Control (IFC_SESSION_CONTROL) Says "Session Operations is a bit field indicating whether to commit or discard any pending changes or confirm previously applied changes." but then the definition doesn't specifiy it as bits, it's just an enum. e.g. the LSB isn't discard/commit etc. Also doesn't include the usual line about what controller/responder should do with unspecified bits.)

Resolution:

Accept in Principle

Resolution Description:

This is an enumeration. The document has been updated accordingly.

Number:

27

Commenter:

Chris Kennedy

Section Number:

6.1

Section Title:

Get Interface List

Description:

Get Interface List (IFC_INTERFACE_ID_LIST) The interface identifiers returned are known as Interface IDs and shall be used by other parameters to uniquely identify an interface until an IFC_INTERFACE_TOPOLOGY queued message is received. Please indicate whether heirarchy changes can re-use the same Interface IDs?)

Resolution:

Accept in Principle

Resolution Description:

Added clarifying text to allow reuse of identifiers.

Number:

29

Commenter:

Chris Kennedy

Section Number:

6.1

Section Title:

Get Interface List

Description:

Assuming IDs can bre re-used, what happens if the heirarchy has changed but the Controller sends a parameter message before receiving the queued message? Potentially the Controller will receive data for the wrong Interface ID? Should all messages using Interface IDs be NACKed after a hierarchy change until the Controller requests a new Get Interface List?)

Resolution:

Accept in Part

Resolution Description:

We have added text explaining that out of band topology changes should not be allowed when a session is active. There is no way of knowing when a queued message has been delivered, so there will be some edge cases where a Controller expects a different topology to that on the Responder.

Number:

30

Commenter:

Peter Newman

Section Number:

6.1

Section Title:

Get Interface List

Description:

Explicitly clarify that each interface ID should only be listed once (I assume that's the case) e.g. if you're modelling a switch with multiple ports (interfaces) in the same VLAN, presumably each port should be listed, and the VLANs?)

Resolution:

Reject

Resolution Description:

The document says "A packed list of unique 32-bit identifiers", the task group believes this is sufficient.

Number:

31

Commenter:

Peter Newman

Section Number:

6.1

Section Title:

Get Interface List

Description:

What happens if we've got an interface 2 and a VLAN 2? The standard doesn't seem to explicitly mention it and RFC 3493 doesn't mention VLANs. In fact "4.3 Sample Interface Hierarchy[Informative]" even includes physical interface 1 and VLAN 1.)

Resolution:

Accept in Principle

Resolution Description:

There is no relationship between and Interface Identifier and a VLAN ID. We have added clarifying text, and revised the informative hierachy diagram to help make this clearer.

Number:

32

Commenter:

Peter Newman

Section Number:

6.1

Section Title:

Get Interface List

Description:

It would probably be good to have a general section on interface IDs specifying that 0 is reserved and also dealing with the max ID, plus the VLAN stuff mentioned above). E1.37-2 had "The interface identifiers returned shall range from 1 to 0xFFFFFF00." so it may be beneficial to track that upper limit across for compatibility". Sadly the E1.37-2 reserved block at the top isn't big enough for all the VLANs, but having a block/mask at the top for the VLANs may be a good solution (so VLAN ID | 0xFFFFF000).)

Resolution:

Reject

Resolution Description:

We have cleared up the association between virtual interfaces, VLANs, interfaces and Interface IDs.

Number:

33

Commenter:

Peter Newman

Section Number:

6.1

Section Title:

Get Interface List

Description:

Add a line that "Interface identifiers are not required to be contiguous.", this both matches the old E1.37-2 spec and allows implementers to just take their native IDs and use them without having to remap them (and risk introducing errors).)

Resolution:

Reject

Resolution Description:

We have slightly improved the text, but the task group felt it was already clear that they may not be contiguous.

Number:

34

Commenter:

Peter Newman

Section Number:

6.2

Section Title:

Get Response Interface Topology

Description:

Get Response Interface Topology (IFC_INTERFACE_TOPOLOGY) The standard says "This parameter is intended for use with queued messages. Controllers shall not issue a GET_COMMAND for this parameter. Responders receiving a GET_COMMAND for this parameter shall NACK with a NACK Reason Code of NR_GET_QUEUED_ONLY." but then later on "Responders receiving a GET_COMMAND shall respond with a GET_COMMAND_RESPONSE with the value of Topology Changed (Bit 0) set to 0.". Which is it? I'm not aware of any other PIDs we can't GET, but can expect a response from, but I might have forgotten one.)

Resolution:

Accept

Resolution Description:

This was an error, the document has been updated to remove the second block of text. It is intentional that this message is only available via queued messages as it has no meaning outside of this context.

Number:

35

Commenter:

Chris Kennedy

Section Number:

6.2

Section Title:

Get Response Interface Topology

Description:

Get Response Interface Topology (IFC_INTERFACE_TOPOLOGY) This queued message response is used for two distinct cases - topology has changed and on initial boot. There is no way to distinguish between the two cases. On boot it is quite normal that the topology will not have changed. A controller that remains permanently on will therefore get notifications of topology change every time the lights are powered on and off.)

Resolution:

Reject

Resolution Description:

Because of the varying lifecycles of responders and controllers it is neccessary for controllers to take the same action at boot and topology change.

Number:

36

Commenter:

Chris Kennedy

Section Number:

6.3

Section Title:

Get Interface Parent

Description:

Get Interface Parent (IFC_INTERFACE_PARENT) Paragraph "Parent ID: The identifier of the parent interface of the interface, or 0 (0x00000000), indicating the interface is on the root." Please add "of the Responder's interface tree" to distinguish from Root sub ids.)

Resolution:

Accept

Resolution Description:

Implemented as suggested.

Number:

37

Commenter:

Peter Newman

Section Number:

6.4

Section Title:

Get/Set Interface Enable

Description:

Get/Set Interface Enable(IFC_INTERFACE_ENABLE) "Some Responders may choose to include Interface IDs for interfaces which are not present in the list of Interface IDs returned from GET:IFC_INTERFACE_ID_LIST. For example: ports which receive SFP (small form-factor pluggable) modules on a network switch. In such cases, a Responder shall return NOT_PRESENT in GET_COMMAND_RESPONSE." Maybe be more explicit that it's SFPs that aren't fitted. Also clarify what should happen (presumably a NACK) if I try and GET an interface that isn't in the IFC_INTERFACE_ID_LIST? What should happen if I SET a not present interface, the same NACK?)

Resolution:

Accept in Principle

Resolution Description:

Added clarifying text on SFPs, and we have added clarification on the appropriate NACK Reason Codes in Interface List.

Number:

38

Commenter:

Peter Newman

Section Number:

6.4

Section Title:

Get/Set Interface Enable

Description:

Add a line that responders may wish to queue an updated state after a bounce when the interface is re-enabled.)

Resolution:

Accept

Resolution Description:

Implemented as suggested.

Number:

39

Commenter:

Chris Kennedy

Section Number:

6.4

Section Title:

Get/Set Interface Enable

Description:

Get/Set Interface Enable (IFC_INTERFACE_ENABLE) Clarify that rolling back a Bounce does not require the Responder to re-perform the Bounce?)

Resolution:

Reject

Resolution Description:

Bounce is a momentary action over time, so there is no possibility of reverting to a momentary action when a restore occurs.

Number:

40

Commenter:

Chris Kennedy

Section Number:

6.4

Section Title:

Get/Set Interface Enable

Description:

Should Time be less than the Commit Time?)

Resolution:

Reject

Resolution Description:

It does not make sense to rollback an interface that has been bounced, so it doesn't matter how these times interact.

Number:

41

Commenter:

Chris Kennedy

Section Number:

6.4

Section Title:

Get/Set Interface Enable

Description:

The sentence "Some Responders may choose to include Interface IDs for interfaces which are not present in the list of Interface IDs returned from GET:IFC_INTERFACE_ID_LIST." can be read to suggest that Responders can reply to IDs that are not in the GET_IFC_INTERFACE_ID_LIST". Reorder as "Some Responders may choose to include Interface IDs in the list of Interface IDs returned from GET:IFC_INTERFACE_ID_LIST for interfaces which are not present.")

Resolution:

Accept

Resolution Description:

Implemented as suggested.

Number:

42

Commenter:

Peter Newman

Section Number:

6.5

Section Title:

Get Interface Hardware Address

Description:

Get Interface Hardware Address (IFC_INTERFACE_HARDWARE_ADDRESS) Mirror the ARP protocol's hardware addresses as the single biggest/most future-proofed list of hardware addresses. i.e. make the hardware address type field 16 bit and use EUI-48 as type 1 and EUI-64 as type 27. This leaves the standard future-proofed to add additional hardware types by just including their lengths in the table and reduces the risk of programmers mistakes mapping from a widely used standard to the otherwise arbitrary assignments used in this standard. As well as the risk of confusion between E1.37-2 which used ARP's hardware types (where EUI-48/MAC is 1) and this standard (where it's zero).)

Resolution:

Accept in Part

Resolution Description:

The group agreed aligning with the IANA Hardware Types found at https://www.iana.org/assignments/arp-parameters/arp-parameters.xhtml makes sense. However the list remains filtered/restricted to EUI48 and EUI64.

Number:

43

Commenter:

Peter Newman

Section Number:

6.5

Section Title:

Get Interface Hardware Address

Description:

Given (from what I can tell) at least some EUI-64 addresses are derived from EUI-48 addresses, its presumably possible for a device to have both, so shouldn't we be able to specify which one we wish to return during the GET request?)

Resolution:

Reject

Resolution Description:

EUI-48 may be derived from EUI-64, but it is not possible for the interface to have multiple.

Number:

44

Commenter:

Peter Newman

Section Number:

6.5

Section Title:

Get Interface Hardware Address

Description:

By having combined the interface hardware address fetching into one PID (compared to E1.37-2's explicit INTERFACE_HARDWARE_ADDRESS_TYPE1 PID) then I don't believe E1.37-5's JSON schema, or at least some of the other alternative PID schemas, have the ability to have different interpretation formats for the same range of bytes depending on either their length or an associated field, which means controllers have to write more special purpose code, or users get a worse experience if e.g. their MAC address doesn't turn up nicely fomatted.)

Resolution:

Reject

Resolution Description:

This is not the first time there is a requirement to intepret how to display variable fields. It also does not future proof this parameter as it limits to EUI-48/64. Controller code is simplified by combining related information into the same PID wherever possible.

Number:

45

Commenter:

Chris Kennedy

Section Number:

6.6

Section Title:

Get Interface Fixed Label

Description:

Get Interface Fixed Label (IFC_INTERFACE_FIXED_LABEL Fixed Label. References section 3.3 which then references RDM 10.1 which says that unless otherwise specified the length is 32 bytes. Change to "The Fixed Label is a text field as defined in Section 3.3 variable up to 227 bytes".)

Resolution:

Accept

Resolution Description:

Implemented as suggested.

Number:

46

Commenter:

Chris Kennedy

Section Number:

6.6

Section Title:

Get Interface Fixed Label

Description:

Would it be better to call it "IFC_INTERFACE_MANUFACTURER_LABEL" rather than "IFC_INTERFACE_FIXED_LABEL" to avoid the confusion between Fixed Label and Fixed Length?)

Resolution:

Accept

Resolution Description:

We have change the name of this PID in the document.

Number:

47

Commenter:

Chris Kennedy

Section Number:

6.7

Section Title:

Get/Set Interface User Assignable Label

Description:

Get/Set Interface User Assignable Label (IFC_INTERFACE_USER_LABEL) Remove "This parameter is intended for user-definable labels." as it repeats the first paragraph.)

Resolution:

Accept

Resolution Description:

Removed as suggested.

Number:

48

Commenter:

Chris Kennedy

Section Number:

6.7

Section Title:

Get/Set Interface User Assignable Label

Description:

User Label. References section 3.3 which then references RDM 10.1 which says that unless otherwise specified the length is 32 bytes. Change to "The User Label is a text field as defined in Section 3.3 variable up to 227 bytes".)

Resolution:

Accept in Principle

Resolution Description:

Implemented as suggested, apart from the length is 225 to allow for the Session ID in the SET.

Number:

49

Commenter:

Peter Newman

Section Number:

6.7

Section Title:

Get/Set Interface User Assignable Label

Description:

Get/Set Interface User Assignable Label(IFC_INTERFACE_USER_LABEL) What if anything should the IFC_INTERFACE_USER_LABEL default to? Blank is potentially unhelpful (as controllers need to do some magic) but matching IFC_INTERFACE_FIXED_LABEL is potentially confusing.)

Resolution:

Accept in Part

Resolution Description:

We've added clarifying text stating that it is possible for this field to be blank.

Number:

50

Commenter:

Peter Newman

Section Number:

6.7

Section Title:

Get/Set Interface User Assignable Label

Description:

Why does SET IFC_INTERFACE_USER_LABEL return the interface ID?)

Resolution:

Reject

Resolution Description:

The Interface ID is a sub-address mechanism, and this was an intentional addition to retain consistency with other standards that have a similar mechanism.

Number:

51

Commenter:

Peter Newman

Section Number:

6.11

Section Title:

Get Interface Duplex Mode

Description:

Get Interface Duplex Mode(IFC_INTERFACE_DUPLEX_MODE) Doesn't allow setting duplex mode.)

Resolution:

Reject

Resolution Description:

This is intended as a status report, there are complexities with auto-negotiation which go beyond the scope of the intended feature-set for this standard.

Number:

52

Commenter:

Peter Newman

Section Number:

6.11

Section Title:

Get Interface Duplex Mode

Description:

I assume this is intended to return the capability of an interface regarding duplex, rather than the state its currently in? You really need both to make a good judgement as to whether it's operating optimally in case something has auto-negotiated to simplex.)

Resolution:

Accept in Part

Resolution Description:

The task group believes simply having the state is sufficient. This PID has been renamed to clarify.

Number:

53

Commenter:

Chris Kennedy

Section Number:

6.12

Section Title:

Get/Set Energy Efficient Ethernet Enable

Description:

Get/Set Energy Efficient Ethernet Enable (IFC_INTERFACE_EEE) How do we know if the interface supports EEE? Should there be a capability? Noting that some ports may support it whilst others do not.)

Resolution:

Reject

Resolution Description:

As with other parts of RDM, the standard response to an unsupported PID would be to NACK it. We expect implementors would do this same in this case. We have added a NACK Reason Code to indicate this.

Number:

54

Commenter:

Chris Kennedy

Section Number:

6.12

Section Title:

Get/Set Energy Efficient Ethernet Enable

Description:

If interface does not support EEE and Controller attempts to set it then should it be NACKed?)

Resolution:

No Action

Resolution Description:

As with other parts of RDM, the standard response to an unsupported PID would be to NACK it. We expect implementors would do this same in this case. We have added a NACK Reason Code to indicate this.

Number:

55

Commenter:

Chris Kennedy

Section Number:

6.13

Section Title:

Get/Set Interface Power Enable

Description:

Get/Set Interface Power Enable (IFC_INTERFACE_POWER_ENABLE) Reference IFC_INTERFACE_POWER_STATUS to determine whether port supports power)

Resolution:

Accept in Principle

Resolution Description:

Added a reference.

Number:

56

Commenter:

Peter Newman

Section Number:

6.13

Section Title:

Get/Set Interface Power Enable

Description:

Could we move IFC_INTERFACE_POWER_STATUS before IFC_INTERFACE_EEE so that we don't have to go searching for it when reading document top to bottom?)

Resolution:

Accept in Principle

Resolution Description:

We have reordered these sections in the document moving EEE lower down.

Number:

57

Commenter:

Peter Newman

Section Number:

6.13

Section Title:

Get/Set Interface Power Enable

Description:

Get/Set Interface Power Enable (IFC_INTERFACE_POWER_ENABLE) It would be good to add the bounce options from IFC_INTERFACE_ENABLE to enable a PoE fed device to be easily bounced, which potentially could be in the control path to the controller.)

Resolution:

Accept

Resolution Description:

We have implemented this as suggested.

Number:

58

Commenter:

Peter Newman

Section Number:

6.14

Section Title:

Get Interface Power Status

Description:

"The reserved value of 0xFFFF shall be reported in the Power Requested, Power Granted, or Power Consumedif the interface does not have the required information available to report." This value isn't defined as a constant in the defined parameters like they normally are.)

Resolution:

Accept

Resolution Description:

Added as suggested.

Number:

59

Commenter:

Peter Newman

Section Number:

6.14

Section Title:

Get Interface Power Status

Description:

"The reserved value of 0xFFFF shall be reported in the Power Requested, Power Granted, or Power Consumed if the interface does not have the required information available to report." Any reason "Maximum Power Capability" isn't covered by this case too? I could imagine some modular device and a third party PoE chipset where it might not be able to read that info either (or varies by hardware) but it at least knows its power sourcing.)

Resolution:

Reject

Resolution Description:

Maximum Power Capability is an immutable attribute of the port, so this has to be available.

Number:

60

Commenter:

Chris Kennedy

Section Number:

6.14

Section Title:

Get Interface Power Status

Description:

Get Interface Power Status (IFC_INTERFACE_POWER_STATUS) Many systems have a total maximum power across all network ports separate to individual port limits which is the most important limit. We can calculate the total used by summing each port, but there does not seem to be a method for reporting the total maximum power.)

Resolution:

Accept in Principle

Resolution Description:

We have added an Available Power Budget field to provide the power remaining across multiple ports.

Number:

61

Commenter:

Chris Kennedy

Section Number:

7

Section Title:

IPv4 Configuration Messages

Description:

IPv4 Configuration Messages It should be allowed to set IP network settings for interfaces that are disabled and may in some cases be desireable to do it before enabling the port. Compare with PC. Arguably also for interfaces that are not available. Could we change to "If the interface is disabled or not present, the Responder may choose to NACK with a NACK Reason Code of NR_INTERFACE_UNAVAILABLE.")

Resolution:

Accept

Resolution Description:

Implemented in doc (in all cases where this appears).

Number:
62

Commenter:
Peter Newman

Section Number:
7.1

Section Title:
Get/Set IPv4 Address

Description:
Get/Set IPv4 Address (IFC_IPV4_ADDRESS) No mention of what to do in ACK_OVERFLOW; presumably drop the interface ID on subsequent packets? Or is it not supported?)

Resolution:
Reject

Resolution Description:
ACK Overflow is only supported for specific parameter messages to assist with support of LLRP.

Number:
63

Commenter:
Peter Newman

Section Number:
7.1

Section Title:
Get/Set IPv4 Address

Description:
If ACK_OVERFLOW isn't supported, SET PDL should be reduced as we can currently SET more than we can read back via GET (due to the 1 byte overhead of source when GETing).)

Resolution:
Accept

Resolution Description:
The document has been updated to make these symmetric.

Number:

64

Commenter:

Peter Newman

Section Number:

7.1

Section Title:

Get/Set IPv4 Address

Description:

If an IP has been assigned via DHCP (for example) does SETting one or more others just set them statically and remove the DHCP assignment? This isn't clarified anywhere. Likewise if I wish to add another "alias"/"virtual" IP I assume I need to SET both the existing and new one to make them both stay?)

Resolution:

Accept in Part

Resolution Description:

We have added some clarifying text to make it clear this is a full set.

Number:

65

Commenter:

Peter Newman

Section Number:

7.2

Section Title:

Get/Set IPv4 Gateway

Description:

Windows, somewhat bafflingly, allows me to set multiple gateways on one interface; which should this return, the first?)

Resolution:

Accept

Resolution Description:

Add text to indicate this might return the default gateway or lowest cost routing metric.

Number:

66

Commenter:

Peter Newman

Section Number:

7.2

Section Title:

Get/Set IPv4 Gateway

Description:

IFC_IPV4_GATEWAY allows me to GET or SET a gateway per interface, but doesn't give me any indication of the routing table on the device and which will win and be the default route.)

Resolution:

Reject

Resolution Description:

This is outside the scope of this document.

Number:

67

Commenter:

Peter Newman

Section Number:

7.2

Section Title:

Get/Set IPv4 Gateway

Description:

Get/Set IPv4 Gateway (IFC_IPV4_GATEWAY) What value should be returned for IPv4 Gateway if it's not set/has been removed with IFC_IPV4_GATEWAY_REMOVE?)

Resolution:

Accept in Principle

Resolution Description:

It is now possible to use this message to remove a gateway by omitting the field. This same mechanism is used to indicate no gateway is assigned.

Number:

68

Commenter:

Peter Newman

Section Number:

7.3

Section Title:

Set Remove IPv4 Gateway

Description:

Set Remove IPv4 Gateway (IFC_IPV4_GATEWAY_REMOVE) Given IFC_IPV4_GATEWAY has to have a way to represent a non-existent gateway in 4 bytes, why not make this PID redundant and just set the magic value to IFC_IPV4_GATEWAY to remove any configured gateway? This would also make that PID more symmetric and remove this non-standard one.)

Resolution:

Accept in Part

Resolution Description:

We have removed this PID, and allow setting of no gateway using a shortened Set:IFC_IPV4_GATEWAY message.

Number:

69

Commenter:

Chris Kennedy

Section Number:

7.3

Section Title:

Set Remove IPv4 Gateway

Description:

Set Remove IPv4 Gateway (IFC_IPV4_GATEWAY_REMOVE) It is unclear why this PID is required rather than using IFC_IPV4_GATEWAY with zero length PDL. For IP addresses there is IFC_IPV4_ADDRESS and no equivalent remove PID. The same comment applies to IFC_IPV6_GATEWAY. Either remove the PID or explain.)

Resolution:

Accept

Resolution Description:

We have removed this PID, and allow setting of no gateway using a shortened Set:IFC_IPV4_GATEWAY message.

Number:

70

Commenter:

Peter Newman

Section Number:

7.4

Section Title:

Get/Set IPv4 DNS Address

Description:

Get/Set IPv4 DNS Address (IFC_IPV4_DNS) On a lot of Linux systems (for example) DNS servers can either be configured at the interface level or centrally, although are processed from the central list, so the interface a server came from may not be easily known (and might not necessarily be directly within its subnet). What should be returned in this situation?)

Resolution:

Accept in Principle

Resolution Description:

We have added some clarifying text indicating this should be related to the resolver associated with the interface.

Number:

71

Commenter:

Peter Newman

Section Number:

7.4

Section Title:

Get/Set IPv4 DNS Address

Description:

"If more IPv4 DNS server addresses are provided than can be accepted, the Responder shall NACK with a NACK Reason Code of NR_IPV4_LIMIT_EXCEEDED." What should happen in the unlikely event the responder has more configured than can be returned (56)?)

Resolution:

Accept in Principle

Resolution Description:

The document has been updated to indicate you shall only return those that can fit within the message.

Number:

72

Commenter:

Chris Kennedy

Section Number:

7.6

Section Title:

Get/Set IPv4 Mode

Description:

Get/Set IPv4 Mode (IFC_IPV4_MODE) The capabilities retrievable with IFC_IPV4_SUPPORTED_MODES are not the same as the modes assignable with IFC_IPV4_MODE. It is likely that many fixtures will support both STATIC and DHCP but will not support fallback. Therefore despite getting IFC_IPV4_SUPPORTED_MODES the set of IFC_IPV4_MODE will fail with NR_MODE_UNSUPPORTED. IFC_IPV4_SUPPORTED_MODES should be modified to provide the full capabilities. Also applies to IFC_IPV6_SUPPORTED_MODES.)

Resolution:

Accept

Resolution Description:

The document now uses the same combinations for both capabilities and set.

Number:

73

Commenter:

Peter Newman

Section Number:

7.7

Section Title:

Get DHCPv4 Server Result

Description:

Get DHCPv4 Server Result (IFC_IPV4_DHCP_RESULT) Change "DHCPv4 Address" to "DHCPv4 Server Address" to avoid confusion with people putting the DHCP assigned address, rather than the server's address in the field.)

Resolution:

Accept

Resolution Description:

We have modified the name to match the description.

Number:

74

Commenter:

Peter Newman

Section Number:

8.5

Section Title:

Get/Set IPv6 DNS Address

Description:

Get/Set IPv6 DNS Address (IFC_IPV6_DNS) PDL or GET Response and SET Request should be "Variable (0x04-0xE4) 0x04 + (Multiple of 0x10)", rather than 0xDC and 0x12.)

Resolution:

Accept

Resolution Description:

Corrected in the document.

Number:

75

Commenter:

Peter Newman

Section Number:

8.8

Section Title:

Get IPv6 Supported Modes

Description:

Get DHCPv6 Server Result (IFC_IPV6_DHCP_RESULT) Change "DHCPv6 Address" to "DHCPv6 Server Address" to avoid confusion with people putting the DHCP assigned address, rather than the server's address in the field.)

Resolution:

Accept

Resolution Description:

We have modified the name to match the description.

Number:

76

Commenter:

Peter Newman

Section Number:

9.1

Section Title:

Get DNS Capability

Description:

Get DNS Capability(IFC_DNS_CAPABILITIES) ""Note: Depending on the Responder's operating system, DNS configuration may only exist on the root device (0x0000), while in other cases it may be per-interface."" I'm unclear why this note is just attached to DNS capability? Surely depending on how the device is ""modelled"" it may only have network interfaces on the root device or on sub-devices. Or is this actually meaning something about particular interfaces rather than sub-devices? It certainly seems more likely for most devices that they will have a single DNS hostname (etc) rather than one per interface. Is this note actually supposed to mean you should return and interact with the reserved root interface (0x00000000) from the hierarchy tree to set the hostname globally? If so that needs a lot more flagging if the root of the tree has a set of properties here but doesn't appear in the main list of interfaces elsewhere?)

Resolution:

Accept in Principle

Resolution Description:

We have added clarifying text and resolved some errors in the original.

Number:

77

Commenter:

Chris Kennedy

Section Number:

9.1

Section Title:

Get DNS Capability

Description:

Get DNS Capability (IFC_DNS_CAPABILITIES) The comment "Note: Depending on the Responder's operating system, DNS configuration may only exist on the root device (0x0000), while in other cases it may be per-interface." Does Root refer to the Root sub device or the root of the Interface tree? The fact that it has a 4 digit 0x0000 sugests Sub Device not Interface ID.)

Resolution:

Accept in Principle

Resolution Description:

We have added clarifying text and resolved some errors in the original.

Number:

78

Commenter:

Peter Newman

Section Number:

9.2

Section Title:

Get/Set DNS Hostname (Label)

Description:

Get/Set DNS Hostname(Label)(IFC_DNS_LABEL) PDL or GET Response and SET Request should be "Variable (0x05 –0x43)", rather than 0x04 as the DNS label has to be a minimum of one character.)

Resolution:

Accept

Resolution Description:

Corrected in document, and added reference to RFC 2181 where this is clarified to 1 octet minimum.

Number:

79

Commenter:

Chris Kennedy

Section Number:

10.3

Section Title:

Get/Set VLAN Label

Description:

Get/Set VLAN Label (IFC_VLAN_LABEL) Label. References section 3.3 which then references RDM 10.1 which says that unless otherwise specified the length is 32 bytes. Change to "The Label is a text field as defined in Section 3.3 variable up to 227 bytes".)

Resolution:

Accept in Principle

Resolution Description:

This has been corrected, however, the field length has changed.

Number:

80

Commenter:

Peter Newman

Section Number:

10.3

Section Title:

Get/Set VLAN Label

Description:

Doesn't this PID clash with IFC_INTERFACE_ID_LIST as that wanted VLANs in the main list, as 32 bit interface IDs, but this has them as 16 bit padded entries of a particular interface ID?)

Resolution:

Reject

Resolution Description:

Interface IDs and VLAN IDs are not the same thing, and the beginning of the document does not claim they are.

Number:

81

Commenter:

Peter Newman

Section Number:

10.3

Section Title:

Get/Set VLAN Label

Description:

"For SET_COMMAND, if the Label string is longer than supported, the Responder shall NACK with a NACK Reason Code of NR_DATA_OUT_OF_RANGE as defined in [RDM]." Pretty certain that particular edge case would be a format error in RDM, whereas trying to set a VLAN label for VLAN 5000 would be data out of range.)

Resolution:

Accept

Resolution Description:

We have changed this to the appropriate NACK Reason Code.

Number:

82

Commenter:

Chris Kennedy

Section Number:

Appendix A

Section Title:

Defined Parameters

Description:

None of the PIDs are marked as required, but section 6 implies that IFC_SESSION_CONTROL is required if other SET_PARAMETERS messages are supported.)

Resolution:

Accept

Resolution Description:

Marked as required when any SET_COMMAND is supported.

Number:

83

Commenter:

Chris Kennedy

Section Number:

Appendix A

Section Title:

Defined Parameters

Description:

IFC_INTERFACE_LIST and IFC_INTERFACE_TOPOLOGY are required if other network parameter messages are supported.)

Resolution:

Accept

Resolution Description:

Marked as required when any other message is supported.

Number:

84

Commenter:

Chris Kennedy

Section Number:

Appendix A

Section Title:

Defined Parameters

Description:

The standard provides no generic way for identifying the function of a network port with respect to the fixture. Nearly all fixtures will have a network port that supports the primary control of the fixture using Art-Net/sACN - but there is no generic way from these PIDs of detecting which port in the hierarchy it is. There should be a numeric Interface Function type defined in the standard with e.g. Primary control port, Secondary control port.)

Resolution:

Reject

Resolution Description:

The task group believes the fixed (manufacturer) and user definable labels are the most appropriate solution for this problem.

Number:

85

Commenter:

Peter Newman

Section Number:

Appendix A

Section Title:

Defined Parameters

Description:

Why is packing universally disallowed? This will make for a very slow way to query a switch type device with lots of interfaces...)

Resolution:

Reject

Resolution Description:

Packing introduces complexity, mainly due to the disallowing of ACK_OVERFLOW in LLRP, and the fact that indexes are 16-bit in contrast with those in this document which are 32-bit. The group also believes the majority use of this document will be outside of RS485 and therefore this isn't a significant issue.

Number:

86

Commenter:

Peter Newman

Section Number:

Table A-2

Section Title:

Defines

Description:

I believe the current SESSION_WINDOW duration is too short for manual use, e.g. someone using a simpler controller where they might be having to manually enter bytes (or even something command line based or similar that does some amount of pretty-printing), anything without a custom UI which handles the session ID management automagically. From what I currently understand (but as mentioned elsewhere some of the session based stuff is very unclear), you'd need to do at least the following (for example, just to make the most basic change without sawing off a branch): GET IFC_SESSION_CONTROL SET IFC_IPV4_ADDRESS SET IFC_IPV4_GATEWAY SET IFC_SESSION_CONTROL COMMIT_RUNNING)

Resolution:

Reject

Resolution Description:

Sessions are not intended for manual use, they are intended to apply a series of changes staged by a Controller programatically.

Number:

87

Commenter:

Peter Newman

Section Number:

Table A-5

Section Title:

Interface States

Description:

The table only lists a description for Set for "BOUNCE", whereas the words for 6.4 say "For GET_COMMAND_RESPONSE, returns whether the interface is enabled, disabled or not present. BOUNCE shall not be returned, unless a previous SET_COMMAND using BOUNCE is still in progress.". Likewise unknown lists a get description, but isn't listed in 6.4.)

Resolution:

Accept

Resolution Description:

The document has been updated to correct these issues.

Number:

88

Commenter:

Peter Newman

Section Number:

Table A-7

Section Title:

Interface Types

Description:

Does SFP fall under "COMBINATION", maybe add it as an explicit example? What about those switches where some ports are shared between say an SFP cage and an Ethernet port? Should one of those be a separate category?)

Resolution:

Reject

Resolution Description:

The task group believes the table is clear. If the SFP is wired (copper) only then it belongs to WIRED, if it can support both it belongs to COMBINATION etc.

Number:

89

Commenter:
Peter Newman

Section Number:
Table A-10

Section Title:
Power States

Description:
First column header is "Interface State" rather than say "Power State")

Resolution:
Accept

Resolution Description:
Corrected.

Number:
90

Commenter:
Peter Newman

Section Number:
Table A-11

Section Title:
IFC Power Type

Description:
First column header is "Interface State" rather than say "IFC Power Type")

Resolution:
-

Resolution Description:
Corrected.

Number:
91

Commenter:

Peter Newman

Section Number:

Table A-12

Section Title:

IFC Power Role

Description:

First column header is "Interface State" rather than say "IFC Power Role")

Resolution:

-

Resolution Description:

Corrected.

Number:

92

Commenter:

Thierry Dupont

Section Number:

Table A-13

Section Title:

IPv4 Sources

Description:

Zeroconf should be explained in the context of E1.37-8. Not all implementors are expert in networking. We should provide examples showing the process that would avoid interpretation. For the less experienced implementor explaining the relationship between DHCP and Zeroconf or BOOTP would be very helpful.)

Resolution:

Accept in Principle

Resolution Description:

We have removed references to ZeroConf and changed these to Link-Local which hs references.

Number:

93

Commenter:

Thierry Dupont

Section Number:

Table A-13

Section Title:

IPv4 Sources

Description:

BOOTP should be explained with examples/scenarios as it is not very known.)

Resolution:

Accept in Principle

Resolution Description:

BOOTP has now been removed from the document so does not need explaining.

Number:

94

Commenter:

Peter Newman

Section Number:

Table A-13

Section Title:

IPv4 Sources

Description:

A value should be added to represent DHCP unknown. As I said during the writing of E1.37-2, "It's very hard on Linux to find out if the current address was assigned by DHCP or not (at least 4 userspace programs on Debian); we can't partially support the PID, but want to expose the core IP data, so what about either splitting DHCP status off to a separate PID, or adding an extra state to DHCP of unknown, for when the software can't tell.")

Resolution:

Accept

Resolution Description:

We have added an unknown option.

Number:

95

Commenter:

Thierry Dupont

Section Number:

Table A-14

Section Title:

IPv4 Supported Modes

Description:

It says: "The interface supports assigning an IPv4 address using some zero-configuration method" The word "some" for an implementor is ambiguous and should be clarified. Are we talking about method listed in Table A-15: IPv4 Modes? We guess it is but we can be wrong.)

Resolution:

Accept in Principle

Resolution Description:

ZeroConf has been removed so this text is now clear.

Number:

96

Commenter:

Thierry Dupont

Section Number:

Table A-15

Section Title:

IPv4 Modes

Description:

We would suggest giving examples/scenarios to explain what the modes are. It might be very clear for a network expert, but it might not be the case for implementors.)

Resolution:

Accept in Principle

Resolution Description:

We have removed some of the lesser known constructs and the table is now clearer using well known modes.

Number:

97

Commenter:

Peter Newman

Section Number:

Table A-15

Section Title:

IPv4 Modes

Description:

""A static IPv4 address assignment shall be used, with no fallback."" and ""An attempt to assign an IPv4 address using zero-configuration shall be performed first, with no fallback."" surely both static and (probably) Zeroconf can't fail (well assuming there's a static address configured in the first case), so the concept of their being a fallback doesn't really make sense? Calling them (and DHCP_NONE and BOOTP_NONE) _ONLY might make this clearer?)

Resolution:

Reject

Resolution Description:

Static can fail in the event of an IP address conflict.

Number:

98

Commenter:

Peter Newman

Section Number:

Table A-19

Section Title:

IPv4 Mode Defines

Description:

First column header is "Method Define" rather than say "IPv6 Mode")

Resolution:

Accept

Resolution Description:

Corrected.

Number:

99

Commenter:

Peter Newman

Section Number:

Table A-22

Section Title:

VLAN Tagged VLAN Flags

Description:

First column header currently mirrors the previous table and should be something more specific, e.g. "VLAN Tagged VLAN")

Resolution:

Accept in Principle

Resolution Description:

Renamed to Tagged VLAN Flags

Number:

100

Commenter:

Peter Newman

Section Number:

Table A-23

Section Title:

Additional Response NACK Reasons Codes

Description:

Whilst admittedly we aren't particularly numerically challenged in terms of NACK codes with 16 bits available and a tiny fraction used, NR_IPV4_LIMIT_EXCEEDED/NR_IPV6_LIMIT_EXCEEDED/NR_VLAN_LIMIT_EXCEEDED do feel rather specific and a bit of a burden for controllers to implement, whilst also not being very flexible for other future uses or manufacturer specific re-usage. Something more generic like NR_TOO_MANY_ENTRIES (or some better wording/naming) would achieve the same thing in a more generic way. It's also not like IPv4 and IPv6 addresses appear in the same PID for example, which would make sense of why these need disambiguating.)

Resolution:

Reject

Resolution Description:

The task group believes more granular and specific NACK Reason Codes help with understanding and troubleshooting.

Number:

101

Commenter:

Peter Newman

Section Number:

Table A-23

Section Title:

Additional Response NACK Reasons Codes

Description:

Similar comment to above, but isn't NR_MODE_UNSUPPORTED just NR_DATA_OUT_OF_RANGE like you'd get if you tried to set an invalid DMX_PERSONALITY for example?)

Resolution:

Reject

Resolution Description:

The task group believes it is clearer to have more discrete NACK Reason Codes.