Summary

- New Doc replacing RFC4122
- Fixing Errata
- New Table of Contents
  - Better clarification on versions 3/4/5
  - Addition of UUIDv6/7/8/Max from Peabody Draft 04
  - Better Test Vectors
  - More Implementation Best Practices
- Better clarify UUIDv3/4/5
- General Doc Update Housekeeping
- Looking forward to Draft 01
The story thus far…

- Work until this point was performed under draft-peabody-dispatch-new-uuid-format-04
  - This Draft added UUIDv6/7/8, Max and many implementation best practices missing from 4122.
- At IETF 114 it was decided to officially adopt this work under a new working group: uuidrev
- It was also decided to merge the work of Peabody Draft 04 with RFC4122 into a new doc:
  - draft-ietf-uuidrev-rfc4122bis
- In addition to this, we must address all erata on RFC4122
New Table of Contents (TOC)

- RFC4122 layout and TOC leaves much to be desired
- During Draft 03 of Peabody document we decided to overhaul the TOC
  - Promote readability
  - Provide Better Grouping of related/need-to-know information
- New TOC (High Level Groupings)
  - Introductions/IETF required language
  - UUID Formatting
  - UUID Layouts
  - UUID Best Practices
  - Misc. IETF Considerations
  - References
  - Appendix. Code
  - Appendix. Test Vectors
- Feedback from around the web liked this new TOC format
- Note: “Algorithms to create UUID” was also removed in favor of defining “implementation best practices”
Fixing RFC4122 Errata

- 13 Errata filed against RF4122 which mostly fall into 2 categories:
  - Grammar:
    - 6665, 6225, 3641, 3476, 1428, 1352, 184
    - Easy to fix
  - Endian and Bit Layout Clarifications:
    - 5560, 4976, 4975, 3970, 3546, 1957
    - Harder to fix but my attempt:
      - Better defining layouts for each version.
      - Further clarification at the start of the document about bit, byte order and numbering
      - Consistent language: Big Endian, Most Significant Bits, “Left-most”
      - More examples via test vectors which show the data in ways other than “large walls of text”
Give v3/4/5 more love

- UUIDv1 is the “starting point” for all discussions about v3/4/5
- Their layouts and algos all reference UUIDv1 logic
- This is confusing and leads many issues
- Solution:
  - Decouple these from UUIDv1
  - UUIDv3/4/5 each get their own block layout with individual field descriptions
  - Group together what we know in best practices sections (after review of many v3/5 libraries)
  - Provide better test vectors
  - Where possible, add extra clarifications glossed over by the original document
General Housekeeping

- Replaced all HTTP links with HTTPS links
- Updated all RFC references to their latest version
- Added Security considerations for SHA1 and MD5
Draft 01 Action Item - Namespaces

- **Problem:**
  - In my review of v3/5 libraries only namespaces used are those defined by RFC4122.
  - Further no text describing HOW we obtained these values in the first place

- **Action:**
  - Describe method for creating new static namespace
  - Use this to define new namespace for IOT and Databases (two very common UUID use cases)
Draft 01 Action Item - Running out of Random

- **Problem:**
  - Interim Meeting discussed some problems around “running out of random”

- **Action:**
  - Add this to relevant section when I have the full context
Draft 01 Action Item - Namespace Registration

- **Problem:**
  - IANA namespace template is from 2004/2005, old RFC for IANA namespace registration is deprecated and replaced by RFC 8141

- **Action:**
  - Update namespace template to match new format
Draft 01 Action Item - v7 and v8 small changes

- Problem:
  - Remove "time-based" constraint from version 8 UUID #124
  - Further clarify v7 field description #125

- Action:
  - #124, no impact, removes something that should have been removed in Draft 04
  - #125, adds further clarification around bit layout of v7 for Millisecond timestamp and forward reference to Counters section
Draft 01 Action Item - Variants/Families and Max UUID

- Problem:
  - Reserve variant for Omni-UUID (Max-UUID) #16

- Action:
  - Add better clarification in Variant table that details the Byte and where Max UUID falls in the current “Unused Variant”
Draft 01 Action Item - ISO and ITU

Problem:
- Once upon a time ISO, ITU, IETF UUID specs were the same (IETF RFC4122 == ITU X.667 == ISO/IEC 9834-8:2005)
  - However, X.667 was updated in 2008 and again in 2012
  - Likewise, ISO/IEC 9834-8 was updated in 2008 and in 2014
  - RFC 4122 has never been updated since 2005

Questions:
- How does IETF’s RFC fit into this picture?
  - When I look at libraries I don’t see anybody mention ITU/ISO specs, they clearly use RFC4122 as the main document.
  - Why did we not update when they updated?
- Who ultimately has authority to “add new UUIDs” like v6/7/8/Max.
  - If anybody, how do we ensure those two other documents are in lock-step with our update since our TOC and contents now deviate greatly?
- Should I add new text from their 2008 and 2012/2014 updates?
  - After a few reads the most significant addition would be usage of UUID for both defined OID layouts
  - I don’t see a good changelog so I would need to DIFF the versions to find other data I could transpose.
Draft 01 Action Item - Standards Track

- **Problem:**
  - RFC4122 was submitted as “Proposed Standard” and never elevated to Internet Standard

- **Questions:**
  - Should we be submitting our final UUID document as internet standard to match the authority of ITU Telecommunication standard and ISO International standard?
    - https://www.rfc-editor.org/standards