## Status

- **Scope:** Main specification for structured email → allow to describe content of email messages in a machine-readable format

- **Updates since IETF 118**
  - Draft got adopted by WG 🎉
  - Some minor fixes/restructuring
  - First Open Source implementation released 🎉 (Roundcube Webmail):
    - [https://structured.email/tools/email_clients.html#webmail](https://structured.email/tools/email_clients.html#webmail)
Structured data representation

● Issues
  ○ Serialization (JSON-LD / Microdata)
    ■ https://github.com/hhappel/draft-happel-structured-email/issues/1
  ○ Placement in MIME messages (embedding; multipart/alternative, …)
    ■ https://github.com/hhappel/draft-happel-structured-email/issues/3

● Status
  ○ General trade-offs discussed at IETF 119
  ○ Currently gathering data about potential backwards compatibility (existing messages; client/library implementations)
  ○ Input appreciated
Structured data representation: Language and serialization

- **Formal language**: RDF (W3C standard)

- **Serialization formats**: JSON-LD, Microdata, Others (RDF-XML, Turtle, …)

- **Options**
  - Focus on JSON-LD (SHOULD? MUST?)
  - Also allow Microdata?
    - Technically different workflow for tools (c.f. https://structured.email/tools/email_senders.html)
    - Anecdotal empirics: used by: Easyjet, HRS, Lufthansa, Ryanair

- See also: https://github.com/hhappel/draft-happel-structured-email/issues/1
Structured data representation: Vocabularies

● Proposal
  ○ Recommend Schema.org if it fits the sender’s use case
  ○ Allow any vocabulary otherwise, similar to MIME body parts
  ○ IANA registry?

● See also: https://github.com/hhappel/draft-happel-structured-email/issues/2
Structured data in messages: Placement

● “Partial representation” - options:
  ○ Embed JSON-LD in text/html SCRIPT-tag
    ■ Pro: already in use; more legacy software-friendly
  ○ Leverage “multipart/related”
    ■ Pro: “clean”
  ○ Corner case: “non representation” (e.g., preemptive vacation notice)

● “Full representation”
  ○ Add “application/ld+json” to “multipart/alternative”
    ■ Pro: “clean”
  ○ Alternative: also embed JSON-LD in text/html SCRIPT-tag
    ■ Pro: More legacy software-friendly

● See also: https://github.com/hhappel/draft-happel-structured-email/issues/3
Referencing MIME content in structured data

- How to reuse MIME media content in structured data?
- Proposal: use “cid:” URIs in JSON-LD IRIs
  - Should already OK under existing specs
  - Implementation guidance may be helpful if structured data is removed from email context
- See also: [https://github.com/hhappel/draft-happel-structured-email/issues/4](https://github.com/hhappel/draft-happel-structured-email/issues/4)

```json
{
   "@context": "http://www.schema.org/",
   "@type": "MusicAlbum",
   "@id": "https://open.spotify.com/album/4F1fUukNCGBD5YjDVioJtu",
   "url": "https://open.spotify.com/album/4F1fUukNCGBD5YjDVioJtu",
   "name": "Semantics",
   "image": "cid:album-cover"
}
```
Referencing structured data in messages (1)

and plump, tender shrimp, she pairs smoked paprika with lemon juice for a bright and earthy edge. She says it serves four but, let’s be real, that’s just a suggestion.

In the recipe notes for Kay Chun’s sheet-pan chopped salad with chicken and a feta topping, you’ll find a lively debate about whether to replace the zucchini with olives. Personally, I’d use both, but then I’m a known maximalist when it comes to sheet-pan meals. Kay’s

● Proposal: use “data-id” of HTML elements to reference JSON-LD entities
  ○ <a data-id=”https://cooking.nyt.com/sheet-pan-chopped-salad” href=”…”>sheet-…</a>
  ○ Con: HTML sanitization issue in MUAs?
  ○ See also: https://github.com/hhappel/draft-happel-structured-email/issues/5

```json
{
  "@context": "http://www.schema.org/",
  "@type": "Recipe",
  "@id": "https://cooking.nyt.com/sheet-pan-chopped-salad",
  "name": "Sheet-Pan Chopped Salad with Chicken",
}
```
Referencing structured data in messages (2)

- How to avoid design inconsistencies in partial representation?

- Proposal: specify hint on extend of presentation in HTML
  - `<a data-id="https://cooking...." data-presentation="reference" href="...">`
Structured data across email messages: Forwarding

● Treatment of structured data on (manual) forwarding
  ○ Keeping might be desired in many use cases (e.g., “share by email”)
  ○ Stripping might be desired in certain use cases
    ■ E.g. privacy issues in “on-representation” cases, where structured data is not obvious from HTML content (e.g., preemptive vacation notice)
  ○ No way to change behaviour of legacy email clients

● Any relation to automatic forwarding?

● See also https://github.com/hhappel/draft-happel-structured-email/issues/6
Structured data across email messages: Replies

- Structured email sent in response to a structured email
  - Based on “potentialAction”

- Capturing reply status
  - “\Answered” flag defined for conventional replies (RFC 9051)
  - Proposal: Adding $AnsweredStructured

- See also: https://github.com/hhappel/draft-happel-structured-email/issues/7
Structured data across email messages: Error replies

- An original sender may not assume that a structured email has been processed by a recipient
- But: if a recipient answers with a structured email response, she may want to be informed if this response is invalid for some reason (in order to avoid the false assumption of a proper reply)

Options
- Define error based on DSN/MDN
- Define structured data error type

See also https://github.com/hhappel/draft-happel-structured-email/issues/8
Structured data across email messages: Updates

- Human readable messages can be recalled/corrected by a follow-up human readable message
- How to recall/update structured information in a machine-readable way?

- Options
  - Use SUPERSEDES header (RFC4021)?

- Potential side issues (answered? forwarded? ...)
- See also [https://github.com/hhappel/draft-happel-structured-email/issues/9](https://github.com/hhappel/draft-happel-structured-email/issues/9)
Processing considerations

- Structured data may be processed on server- or client-side
  - Similar challenges in Calendaring (iMIP) or MDN
- Machine-derived structured data
  - Some ISPs and tools derive structured data from semi-structured content (text, images, attachments)
- Metadata helpful to optimize/enable processing
  - $hasStructuredData (Similar to $hasAttachments)
- Might mostly be solved by defining appropriate flags
  - See also: https://github.com/hhappel/draft-happel-structured-email/issues/10
  - Larger question: storing/retrieving structured data independent of the message store?
Action considerations

● “Conventional” emails can contain “materialized actions” (links); e.g.:
  ○ “Geo:-27.47,153.01” (URI scheme)
  ○ https://www.google.com/maps/place/Brisbane+Convention+%26+Exhibition+Centre/(…)

● Structured data provides abstract representation
  ○ Allows for different materializations

● Observation: missing or inadequate URI schemes for many common use cases
  ○ E.g. “spotify://” but not “music://”

```json
{
  "@context": "https://schema.org",
  "@type": "Place",
  "geo": {
    "@type": "GeoCoordinates",
    "latitude": "-27.47",
    "longitude": "153.01"
  },
  "name": "Brisbane C&E Centre"
}
```
Next steps

- Further topics?

- Base questions on structured data representation and placement should be sufficiently clear
  - Some debate on existing options
  - Additional vendor feedback might be helpful
  - Decide until/at IETF 120?

- Issues relation to identifiers and particularly structured data across messages
  - Need feedback and input