

Replies event format

A) Embed in body & formatted_body, putting the eventId as one of the props on one of the tags for the HTML parser in react-sdk to consume and replace with a <Quote />.

Pros:

- Partial (yet ugly) function with clients that do not yet support quoting
- Similar to <Poll />s
- Automatically notifies people prior in the reply chain because their messages (and MXIDs) are embedded within every consequent

Cons:

- Requires parsing of HTML to recognise the reply.
- Does not limit there to be one event being replied to which could yield spam.
- Does not play well over the matrix-appservice-irc bridge where IRC users have expressed their unhappiness about it.
- Event body starts getting heavy after a few replies (hard to strip out previous replies before replying again)
- Does not play well with redactions: the embedded original text remains there even if the original event gets redacted

Example:

```
{
  "origin_server_ts": 1516320312662,
  "sender": "@webdevguru:matrix.org",
  "event_id": "$1516320312749182Zykpw:matrix.org",
  "unsigned": {
    "age": 133449286
  },
  "content": {
    "body": "Wizard t3chguy:\n> Wizard t3chguy:\n> Wizard t3chguy:\n> 1\n\n4\n\n5\n\n123",
    "msgtype": "m.text",
    "formatted_body": "<a
href=\"https://matrix.to/#/!StHFYGUIZlrePtNLpE:matrix.org/$1516306321489604DFZOw:mat
rix.org\">Quote<br></a>123",
    "format": "org.matrix.custom.html"
  },
  "type": "m.room.message",
  "room_id": "!StHFYGUIZlrePtNLpE:matrix.org"
}
```

B) For `m.room.message` events define an additional content field, `in\_reply\_to` which has the eventId of the event being replied to.

Pros:

- Means riot can do its linear-replies thing without having a horrendous recursive context mess.
- Much easier to consume by clients.
- Means the matrix-appservice-irc bridge can choose not to send the reply over the line
- Means that server support could be added to treat replies as their own notifications type
- Lighter payload (significantly over highly nested replies)
- Limits it to a single event being replied to which is the intent.
- Can be extended towards `in\_reply\_to` becoming an object with additional attributes, such as `room` (addressing crossroom quoting), or `collapse` (to trigger showing the original text initially collapsed in the client).

Cons:

- Requires explicit support from clients

Example:

```
{
  "origin_server_ts": 1516320312662,
  "sender": "@webdevguru:matrix.org",
  "event_id": "$1516320312749182Zykpw:matrix.org",
  "unsigned": {
    "age": 133449286
  },
  "content": {
    "body": "123",
    "msgtype": "m.text",
    "in_reply_to": {
      "event_id": "$1516306321489604DFZOw:matrix.org"
    }
  },
  "type": "m.room.message",
  "room_id": "!StHFYGUiZlrePtNLpE:matrix.org"
}
```

Notes:

`in\_reply\_to` is an Object so that it may later be easily extended without causing backwards compatibility issues. Potential things to add later are room_id for cross-room replies and/or ranges for replies to parts of events.

For Replies, B was chosen, but to be implemented in a much more generic relationship manner which can later be implemented as threading server-side.

Continued at

<https://docs.google.com/document/d/1FZsvodn2C0iKJDtn-8y8IPwOa96ixoJeyJ3gMLVOXHM/edit>