

IETF 117 RegExt Session, San Francisco, Jul. 25, 2023



# About objections to JSContact

Mario Loffredo  
Registro.IT/IIT-CNR

# Main objections to JSContact



- *It is too complex for RDAP*
- *It is easy to get wrong*
- *It should use arrays instead of objects/maps*
- *It shouldn't use PatchObject to handle localizations*

# JSContact is too complex for RDAP



- JSContact represents the same information as jCard
- If RDAP implementers haven't been concerned about inapplicable jCard elements, will they be concerned about their JSContact counterparts?
- Unlike jCard, JSContact implementations in RDAP are supported by `calext-jscontact-vcard` and `regext-rdap-jscontact`
- JSContact is a multi-purpose contact representation
- RDAP implementers could benefit from leveraging JSContact libraries developed outside the RDAP ecosystem

# JSContact is easy to get wrong



- JSContact is highly structured and controlled
- In general, the more structured and controlled a data model is, the less ambiguous it is, the more difficult it is to get wrong
- It's unlikely to build wrong objects through a library fully supporting JSContact implementation and validation (e.g. jscontact-tools)

# JSContact should use arrays instead of objects/maps



- The object granularity has been tuned to make objects unambiguous, flexible, extensible and validatable and to ensure full compatibility with jCard
- JSContact uses maps for unordered collections (e.g. emails) and arrays for collections that might be ordered (e.g. address components)
- When map keys are known in advance (as defined by rdap-jscontact):
  - getting a map entry is easier than looping on an array
  - maps can be deserialized into objects having members of the same type (unknown entries can be stored into a map)

# JSContact shouldn't use PatchObject



- PatchObject allows to:
  - localize both simple and complex values
  - use the same process to get localizations of values, arrays, objects (including the card itself) and extensions
  - separate the core information provided in the default language from its localizations
- Is it more efficient than replicating the language information whenever it's needed?



# My conclusions



- Being a format fully compatible with jCard, the amount of information JSContact can represent far outweighs the needs of RDAP but I consider it as a benefit, not a drawback
- As opposed to jCard, JSContact is highly structured and controlled. That said, it's unclear to me how it can be "*easy to get wrong*"
- JSContact structure has been tuned over time to achieve clearness, flexibility and extensibility. Given that, the discussion about using arrays instead of maps looks very simplistic to me
- PatchObject is a flexible, extensible and definitive method to handle any localization. Any alternative solution should be just as efficient.

**Thanks for your attention**

