

# JMAP Sieve

draft-murchison-jmap-sieve-01

Ken Murchison

# What is it?

- A JMAP-based protocol for managing Sieve scripts
- The current design maps ManageSieve (RFC 5804) functionality onto JMAP

# Current design

- The `accountCapabilities` object includes a list of the supported Sieve extensions and any other pertinent information
- A `SieveScript` object represents a single script with the following properties:
  - `id` (`Id`): The server-set, immutable id of the script
  - `Name` (`String`): The user-visible name for the script
  - `content` (`String`): The Sieve code in the script
  - `isActive` (`boolean`): Is this the user's active script?

# Current design

- SieveScript/get: Provides similar functionality to the GETSCRIPT and LISTSCRIPT (if ids argument is null) commands
- SieveScript/set: Provides similar functionality to the PUTSCRIPT, DELETESCRIPT, RENAMESCRIPT, and SETACTIVE commands
- SieveScript/validate: Provides similar functionality to the CHECKSCRIPT command

# Possible additions/changes

- Do we need HAVESPACE functionality?
- SieveScript/query: Filter and/or sort on name and/or isActive
- SieveScript/test: Takes a script id and an email id and returns the set of actions that would be taken
- Use blobIds to reference script content rather than using a string argument?

# Possible additions/changes

- Only allow isActive to be [un]set outside of SieveScript/set{create | update}
  - Add a onSuccessActivateScript argument to SieveScript/set?
  - Use a SieveConfig/set{update} method?

# Next Steps

- Is there enough interest to adopt this into a WG?
- If yes, does this belong in EXTRA or JMAP?