Experimental/private join attributes

Current join attribute registry

- 64 code points, used 4 so far
  - Explicit RPF and LISP join attributes use 3 more.
- Allocation based on IETF review
  - RFC needed (I believe), but may get early allocation when document is stable
- Fairly strict, no code points available for experiments or private use
- Make it more liberal?
  - Concerns with supporting attributes that have not had sufficient review?

Experimental code points?

- What if someone wants to experiment with a new implementation/protocol requiring a new code point?
- RFC 3692 recommends having at least one value for experiments
  - For experiments only, should not be used in products. Different experiments are likely to use the same value, hence not for general deployments.
  - Should be ignored by default

Private code points?

- Set aside some attributes for more private use?
  - Less strict, e.g. require documentation?
  - Registry needed to avoid conflicts?
  - Expand the type space?
- Or do we believe that attributes should be reviewed?
- If we allow private code points, would anyone ask for code points that require review?

Encoded-Source Address Encoding Type Field registry

- 256 values
- 0 is native, 1 for join attributes.
- Allocation based on IETF review

- Fairly strict, any need for experimental or private use?

draft-atwood-pim-reserve-exp-00

- New draft submitted this Monday
- Proposes 2 experimental attribute code points and 4 experimental encondig types

- Is this what we want?