

# Common YANG Data Types

## draft-schoenw-netmod-rfc6991-bis-01

Jürgen Schönwälder  
Jacobs University Bremen

# Additions since RFC 6991

- ietf-yang-types:
  - date, time
  - hours, minutes, seconds
  - centiseconds, milliseconds, microseconds, nanoseconds
  - revision-identifier, node-instance-identifier
- ietf-inet-types:
  - email-address

# Discussion: data / time types

- canonical format of date
  - Need to take a closer look how our canonical format of a 24 hour interval aligns with other date definitions
- time resolution - 32 vs 64 bits
  - The hours, minutes, seconds, centiseconds, milliseconds, microseconds, nanoseconds types currently all use uint32
  - High resolution definitions should likely use uint64; where do we draw the line and move from 32 to 64 bits?

# Discussion: xpath type

- xpath and non-XML encodings
  - How do we deal with xpath expressions in other encodings such as JSON. Do we assume an xpath context populated with module names such that module names can be used to qualify path expressions. This may need discussion and/or a new definition.
  - This interacts with the definition of node-instance-identifier
  - This interacts with yang-next issues #55 and #56

# Discussion: node-instance-identifier type

- node-instance-identifier
  - This is taken from RFC 8341, the idea is that this definition is useful without requiring a dependency on NACM
  - The second bullet in the NACM definition says:

The set of variable bindings contains one variable, 'USER', which contains the name of the user of the current session.

How is this actually used? Do we keep this?

- How does this work with JSON? Can we make this encoding neutral (but then we knowingly depart from NACM)?
- This interacts with the definition of xpath1.0

# Discussion: email address type

- email-address
  - It was suggested to add email types following RFC 5322
    - email-address (addr-spec, per Section 3.4.1)
    - named-email-address (name-addr, per Section 3.4)
  - name-addr is a quite complex construct; perhaps addr-spec is sufficient, this is also the format allowed in mailto: URIs (mailto: seems to use only a subset of addr-spec which may be good enough here as well)
  - Need to define a pattern that has a meaningful trade-off between precision and complexity (there are very tight pattern that are very long and complex)
  - The current pattern does not take care of quoted-string, obs-local-part, domain-literal, obs-domain

# Discussion: additional suggestions

- longitude, latitude, ...
  - It was suggested to add types for longitude, latitude, postal code, country-code. Do we go there or do we leave these for other modules to define?
- percentage types
  - It was suggested to add percentage types but they tend to differ widely. However, percentages are also used widely.