

# Transport Properties

To registry, or not to registry?

# Transport Properties

- Central concept to configure the transport system
- Classified by object / connection phase they effect:
  - Selection Properties (Preconnection)
  - Connection Properties (Connection)
  - Message Properties (Message Context)

but can be specified in earlier phases

- Different data types:
  - ◉ Boolean, Integer, Enumeration, Preference

▶ **Individual Properties are only referenced by section title**

# Should we Standardise Property Names?

## **Pro:**

- “interoperability”: TAPS implementations will use the same names for well-know properties
  - ▶ Key reason for implementing TAPS instead of “something propriety”

## **Contra:**

- Concrete values in an abstract API
- Limits Implementors’ freedom

# Property Registry – Why?

- WG discussions suggest generic transport properties specified in draft-ietf-taps-interface are not conclusive
  - Consensus Properties in Sections 5.2, 7.3, and 9.1
  - Non-Consensus Properties in Appendix A
  - Open debate: All mandatory to implement?
- We need to add protocol-specific properties e.g. for configuring TCP keep-alive
- We need extensibility for new transport features, protocols, and vendors

# Questions

- Should we Standardise Property Names?
- Should we request a registry for Transport Property names?
- What should go into the Registry?
- What assignment policy to use?

# Proposal: Property Names

- Transport Property names are CamelCased strings
- Format: [ <namespace> ( . | \_ ) ] <property name>
- Namespace distinguishes generic/well known from protocol/vendor specific/experimental properties

## Examples:

- ReliableDataTransfer (generic well known property)
- TCP.KeepAlive (protocol specific property)
- Linux.NoRecvMMAP (vendor specific property)

# Proposal: Registries

- Transport Property Namespaces

String	Description	Reference
-----	-----	-----
X	Experimental and Private use	

- Assignment policy: IESG approval
- Reference must point to list of properties in this namespace and assignment policy (external to IETF for vendors and non-IETF protocols)

# Proposal: Registries

- Well-Known Generic Transport Properties

String	Type	Description	Reference
-----	-----	-----	-----

- Assignment policy: RFC required

- Experimental and Private Use Transport Properties

String	Type	Description	Reference
-----	-----	-----	-----

- Assignment policy: Specification Required