

# The MODERN Protocols

Jon Peterson

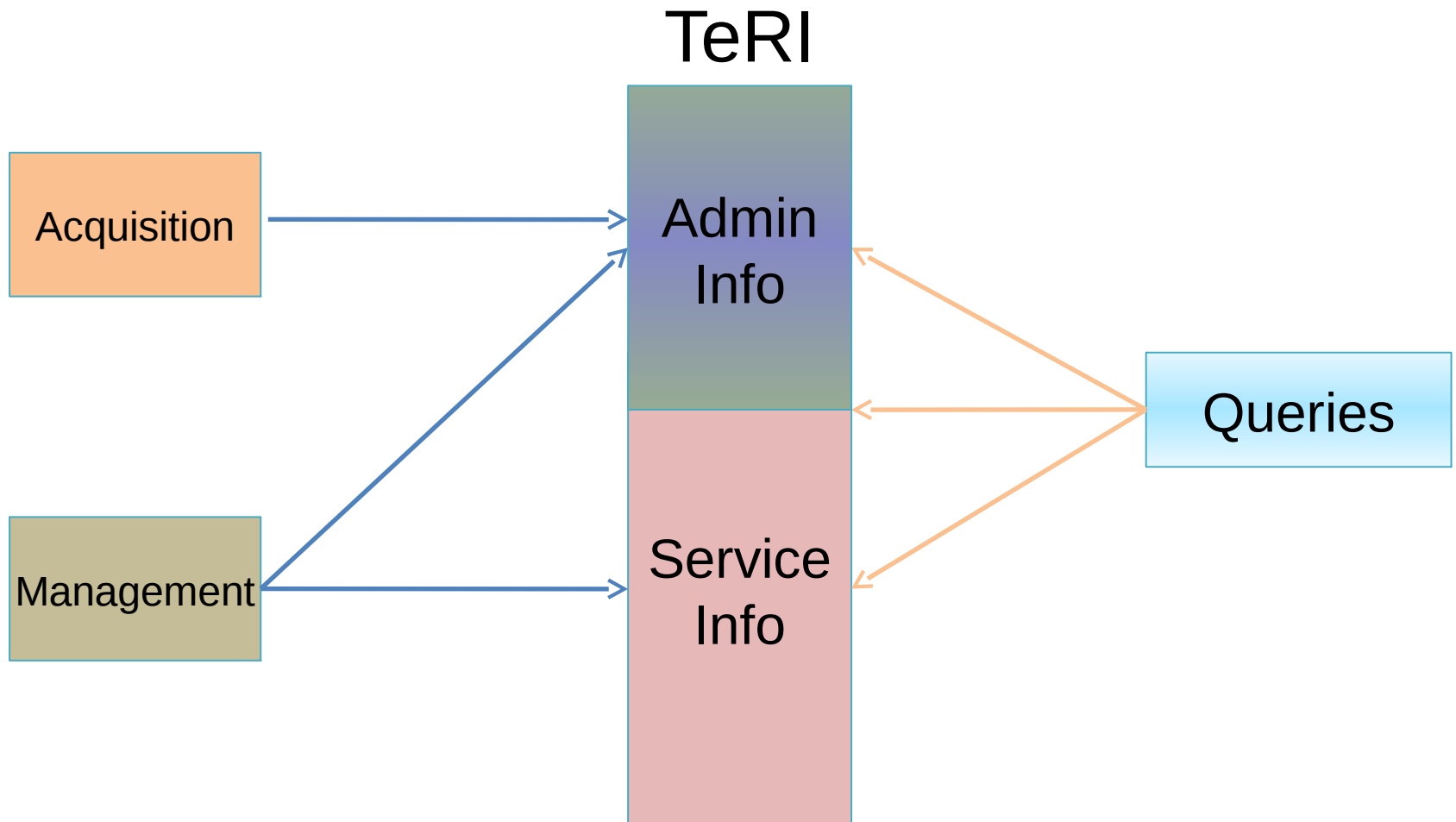
MODERN WG

IETF 93 (Prague)

# Moving Parts

- Acquisition protocol
  - How do I request and receive numbers?
- Management protocol
  - How do I provision a number with a CSP?
- Query protocol
  - How do I get information about a number?
- These protocols access overlapping data
  - If you can provision it, you should be able to query for it
- Surely this is a common information model?

# Telephone-Related Information



Just a logical picture

# Mapping the Model to an Instance

- TeRI Records would live in servers
  - Could be public, centralized and monolithic
  - Could be distributed, or private
  - This logical architecture will be the same
  - Each TN might have multiple Records
- All sorts of entities might manage or query
  - Could be carriers, enterprises, or end users
  - Query access will vary depending on who is asking
  - Provisioning will reflect who provisioned

# Operations and Records

- Proposal: we define all three protocols in terms of this TeRI model
- Each protocol will have its own Operations, but will operate on a common class of TeRI Records
- Operations (Query, Response, etc.) will have their own Source, Subject, and Attributes
  - Source indicates the originator of the Operation
  - Subject would typically be a TN itself (or a range)
- TeRI Records contain information about TNs
  - Some Records might cover a range of TNs

# TeRI Record Contents

- TeRI Records would contain
  - Authority (Source of the data)
  - Elements (Name/Value pairs, embedded Elements(?))
  - Expiration (optional)
  - Priority (optional)
- Divided into Service and Administrative Information
  - This is a distinction we need to explore more
  - Different requirements for returning Service information?
- Obviously different actors would set/get different Records

# Transport and Encoding

- Agree on semantics first, then define bindings and profiles
  - A binding is defined as an encoding and a transport
    - We want at least one binding per protocol, maybe allow more
  - Could build on JSON/HTTP, could build on ASN.1/UDP
  - Bindings need to detail how the elements of the data model are mapped to the encoding
    - Other low-level details like chunking, representation of cryptographic security, etc.
  - Requirement: to transcode between bindings without losing data (at an intermediary)
- Aim for maximum applicability
  - While not overcomplicating the model

# The TeRI Suite

- Core TeRI model
- TeRA
- TeRM
- TeRQ (which I'll talk about shortly!)
  
- Ontologically and organizationally, is this a reasonable plan?
- Where should the core TeRI model live?



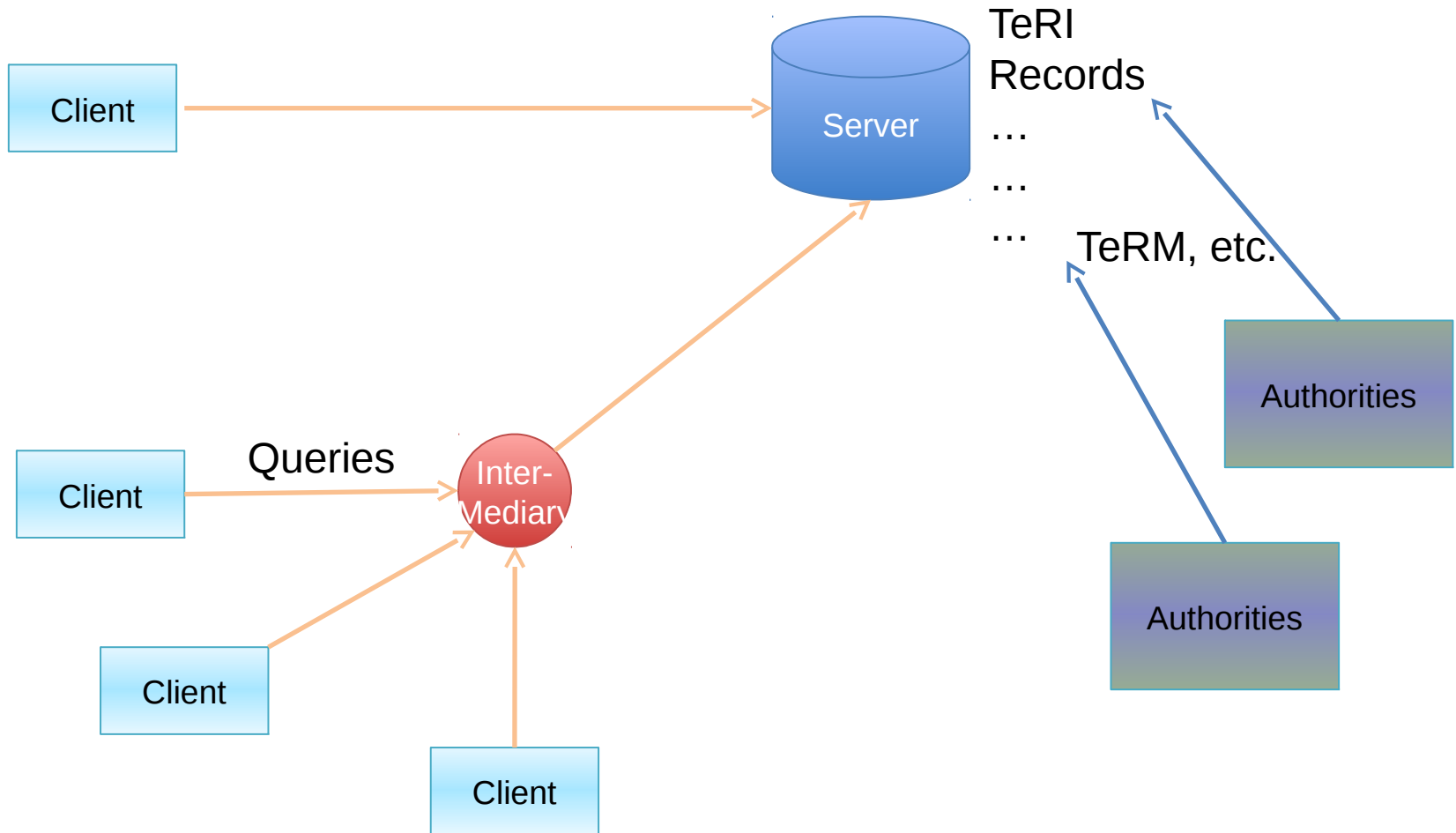
draft-peterson-terq

MODERN WG  
IETF 93 (Prague)

# Why Present TeRQ Today?

- Well, because it's there
- Effectively we will have to explore querying and provisioning/management simultaneously
- Doesn't matter what order we turn these things in to the IESG
  - Could be more of a bundle than a long tail of deliverables for the WG
- We can probably get a sense of how the TeRI design model would work

# The TeRQ Architecture



# TeRQ Operations

- Query:
  - Source (Query Source, Query Intermediary, Route Source)
  - Subject (Telephone Number/Range)
    - Used to have SPID, currently removed per MODERN scope
  - Attributes (constrains query: e.g., “voip” if only looking for VoIP)
- Response:
  - Response Code
  - Subject (Optional)
  - Records (TeRI)

# TeRM Operations (hypothetical)

- Push:
  - Source (Admin Source, Admin Intermediary)
  - Subject (Telephone Number/Range)
  - Record (TeRI)
    - Or individual Element in an existing Record
  - Batching?
- Response:
  - Response Code
  - Subject (Optional)

# TerQ Base Element Types

- Data model current specifies:
  - Telephone Number (RFC3966 – but should we revisit?)
    - Ranges – need some work here
  - Domain Name
  - URI
  - IP Address
    - IPv4/IPv6
  - SPID
    - Currently specified as four-digits, other SPID types possible
      - GSPID, ITAD, etc.
  - Trunk Group
    - Currently points to the Gurbani/Jennings RFC
  - Display Name
    - Support for CNAM as well as a SIP “From” header field
  - Extension
    - Reserved for further use

# TeRQ Profiles

- We anticipate different environments will want different Elements in Records
  - Possibly even different Bindings
    - In HVE, might want very lightweight transports
- Obviously TeRI Records must be extensible
  - Data models for MODERN protocols must similarly allow for that extensibility
- Our mission: start simple
  - But don't paint ourselves in

# Next Steps

- If we are going for a TeRI model, move out the Record information from TeRQ into another spec
- Make more progress