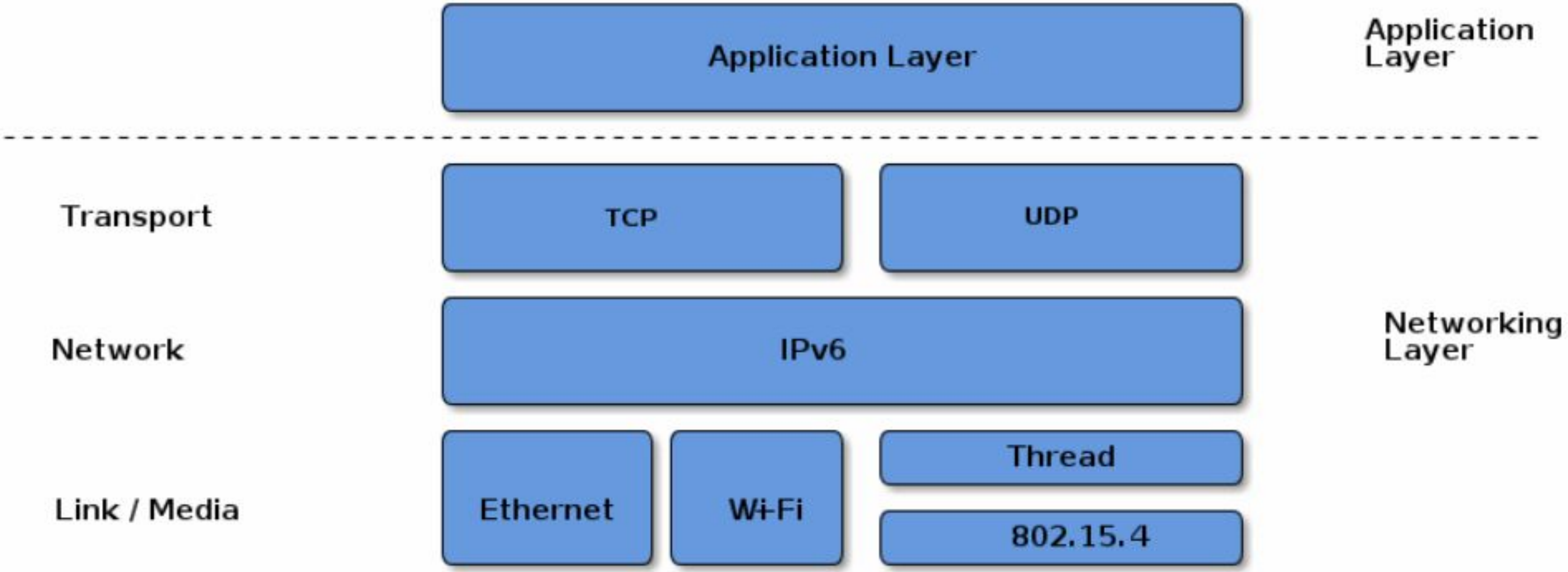
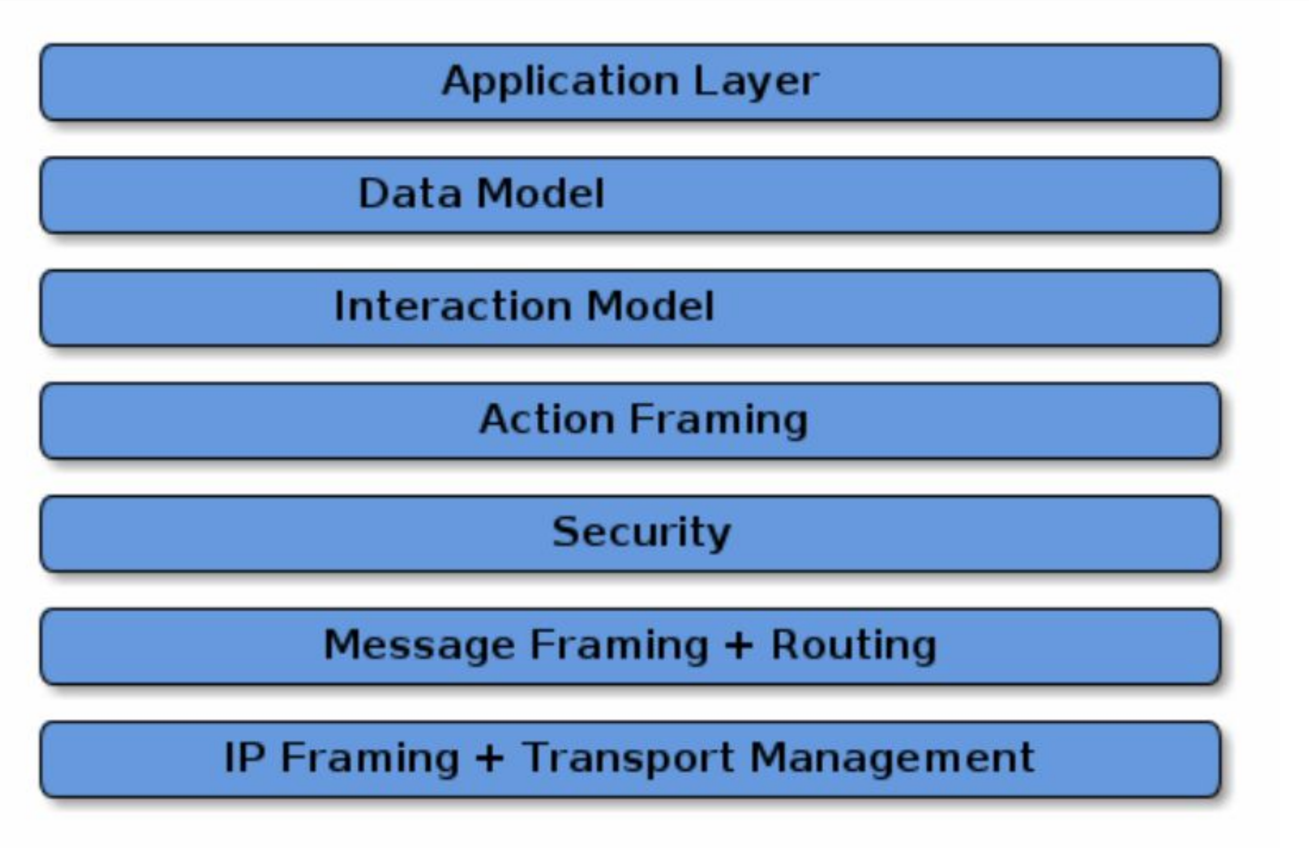
Matter and Open Screen Protocol

W3C Second Screen Working Group (TPAC 2023)
Mark A. Foltz (mfoltz@google.com)
September 15, 2023

Matter Architecture





Application Layer

Data Model

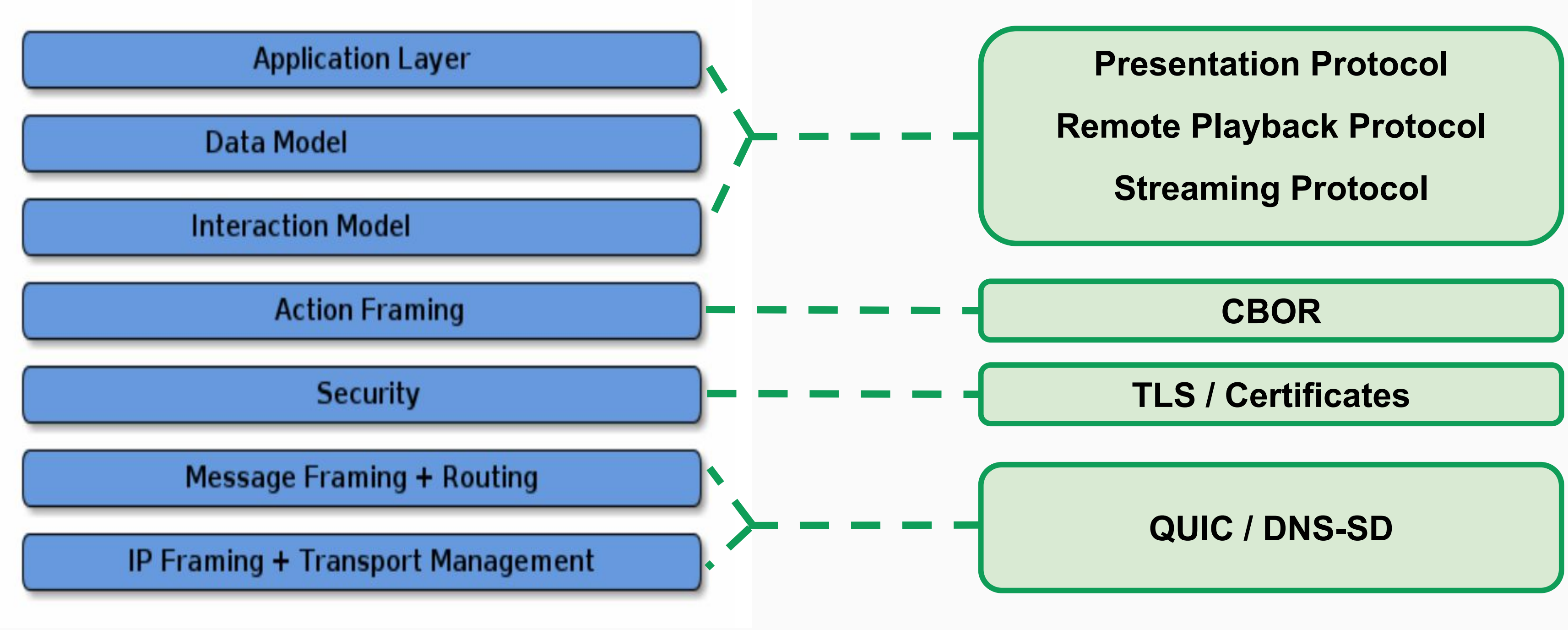
Interaction Model

Action Framing

Security

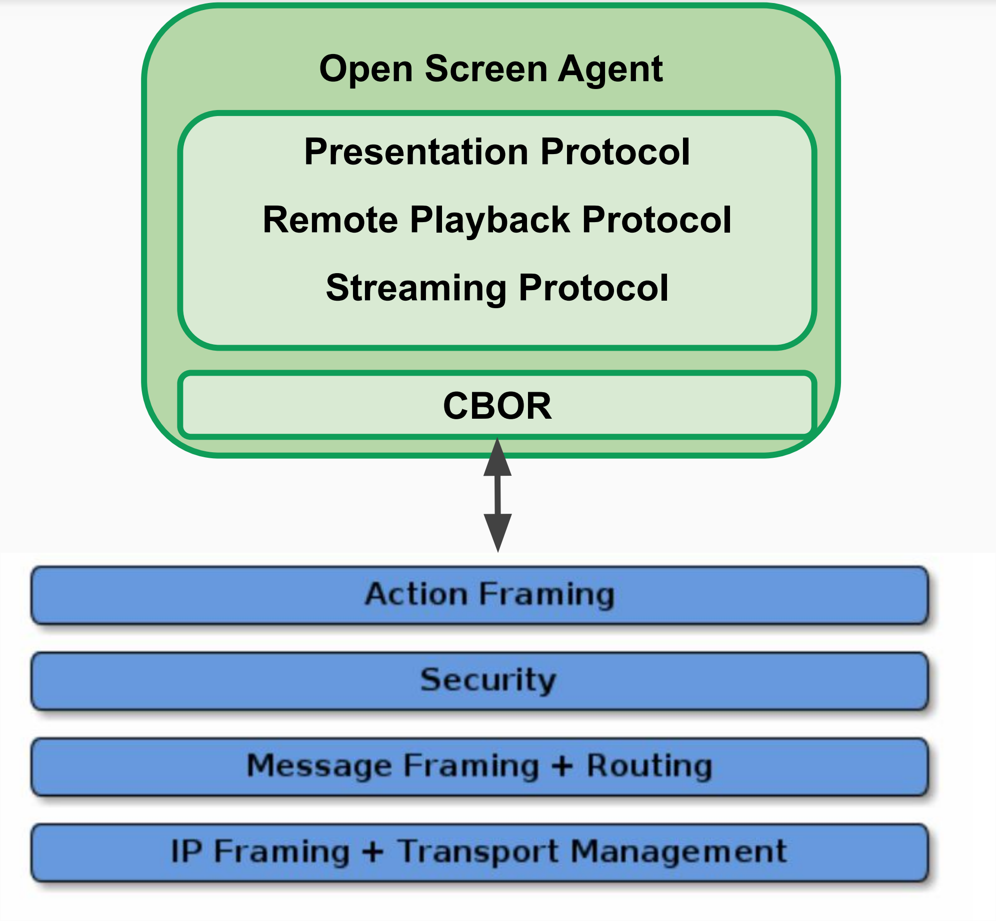
Message Framing + Routing

IP Framing + Transport Management



Matter	Open Screen Protocol
Application Layer Data Model Interaction Model	Presentation Protocol Remote Playback Protocol Streaming Protocol
Action Framing	CBOR
Security	TLS / Certificates
Message Framing + Routing IP Framing + Transport Management	QUIC / DNS-SD

Hybrid approach #1, the “tunneling option”



Open Screen Agent

Presentation Protocol

Remote Playback Protocol

Streaming Protocol

CBOR

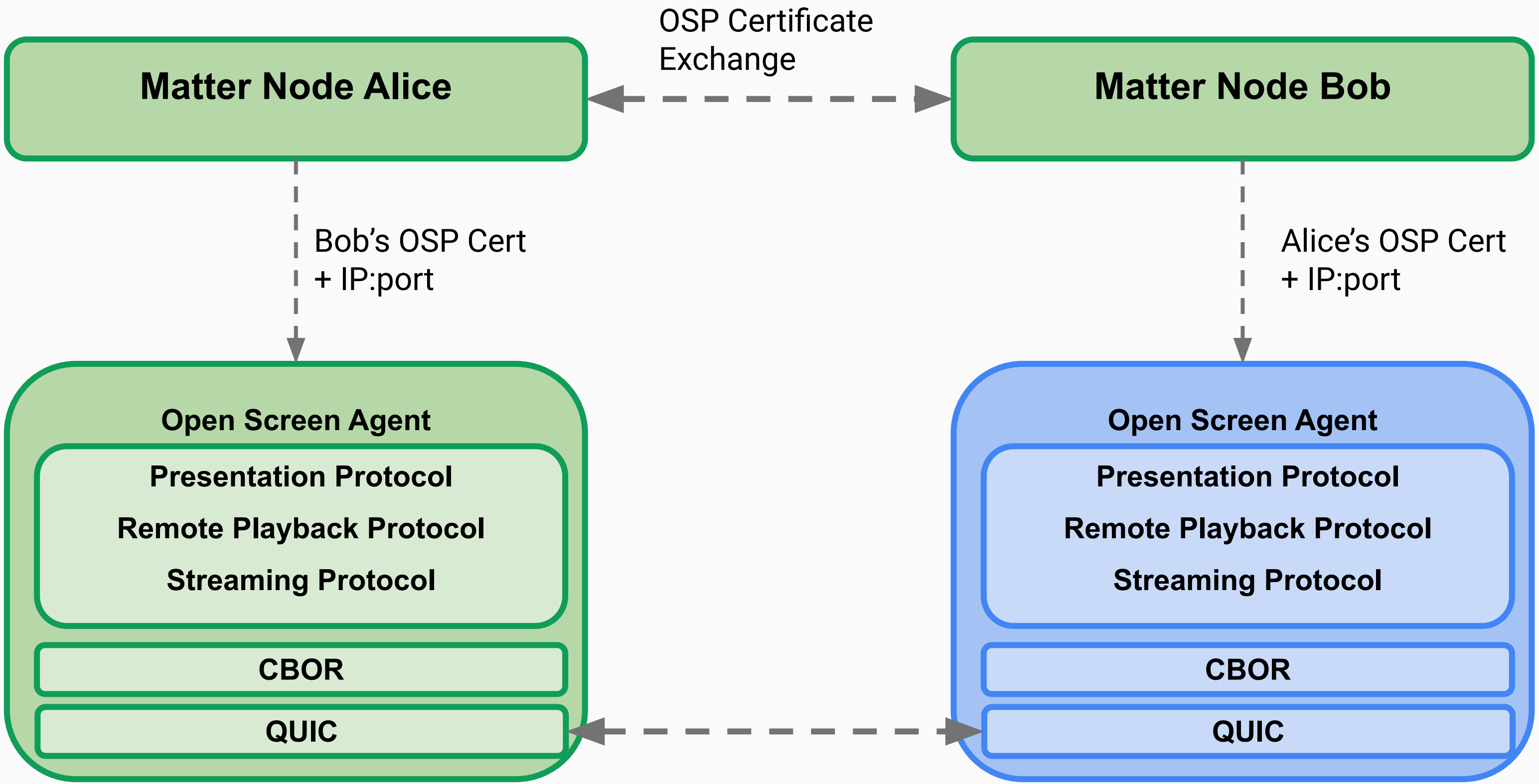
Action Framing

Security

Message Framing + Routing

IP Framing + Transport Management

Hybrid approach #2, the “bootstrap” option



OSP Certificate Exchange

Matter Node Alice

Matter Node Bob

Bob’s OSP Cert + IP:port

Alice’s OSP Cert + IP:port

Open Screen Agent

Presentation Protocol

Remote Playback Protocol

Streaming Protocol

CBOR

QUIC

Open Screen Agent

Presentation Protocol

Remote Playback Protocol

Streaming Protocol

CBOR

QUIC

Areas Of Investigation (Updated)

1. How to use Matter transport to convey CBOR messages (for “tunneling”)?
2. **Suitability of the Matter transport for streaming use cases?**
3. **How to leverage Matter to create an authenticated QUIC connection?**
4. Investigate the “Casting Video Client” device type (Device Library 10.6).

Awaiting progress in CSA on the relevant platform pieces.

Open Screen Protocol 1.0 Spec status (updated)

Label/Category	**Number**	**With PR**
v1-spec	8	8
security-tracker	7	4
privacy-tracker	1	1
meta	1	n/a
Total	**14**	**13**

Open Screen Library implementation progress

Wei Wang (Intel) has been making numerous contributions to the library - thanks Wei!

- mDNS discovery uses our native mDNS module
- QUIC support uses the same implementation as Chrome (QUICHE)
- TODO: Update CBOR messages
- TODO: Implement authentication
- TODO: APIs for remote playback, streaming protocols

Splitting the OSP specification?

The OSP spec has two main parts:

1. Finding, connecting, authenticating between agents.
2. Messages to support the relevant APIs (Remote Playback, Presentation, Streaming)

It would be simpler to write this as two different specs so we could support different solutions for #1.

Why split the OSP specification?

There are multiple ways to create an authenticated connection.

- QUIC + TLS + SPAKE-2 (what OSP requires now)
- Matter + QUIC?
- WebTransport?
- RtcDataChannel, HTTP/3, etc.

It’s possible to exchange OSP messages with all of them.

WebTransport/RtcDataChannel/http

These are interesting because they are already part of the Web, so any browser that supports them would be able to exchange OSP messages.

The messaging part of OSP can be implemented in script; cbor-web is one JS parser for CBOR.

To support encoding/decoding media, you would need WebCodecs.

Possible split?

	Connection	Messaging
API and Non-functional Requirements	✅	✅
Discovery with mDNS	✅	
Transport and Metadata with QUIC	✅	✅
Message Delivery using CBOR and QUIC streams	✅	✅
Authentication	✅	
Presentation API		✅
Remote Playback		✅
Streaming		✅
Security and Privacy, etc.	✅	✅

Recommendations for SSWG (Updated)

1. Land PRs for security-tracker issues in OSP.
2. Land PRs for v1-spec issues.
3. Draft “spec split” if supported by group.
4. ~~Propose a white paper for the “tunneling” approach.~~
5. Draft a protocol extension for the “bootstrap” approach once Matter support is available.

End