Microsoft

April 17-18, 2024
sambaXP Conference

Microsoft Interoperability Track

Microsoft

Exchange Server Protocols Overview

Andrew Davidoff
Senior Software Engineer

Agenda

- Open Specifications Introduction
- Exchange Protocols scope
- Exchange Protocols details
- Choice of protocols to use
- Resources

OPEN SPECIFICATION: BRIEF INTRODUCTION

Introduction to Open Specifications

- Normative language
  - Designed to document bits over the wire
  - No APIs – only Protocols
  - MUST, SHOULD, MAY and endnotes are used to document version-specific behavior. Follows RFC 2119.
  - Strict document structure
- Windows/Exchange or product API knowledge not required…
  - Microsoft Products are not mentioned in normative content, only in endnotes and Overview documents
- Open Specifications document on-premises Exchange Server protocols

Open Specifications Highlights

- Endpoints
  - Most Exchange documents focus on Server endpoint (Server behavior)
  - Some documents prescribe Client endpoint behavior quite extensively (Outlook)
- Product Versions
  - Implementation choice for different releases (product versions) is stated in the Appendix using endnotes
- Example: Endnote clarifying version-specific behavior

2.2.1.56.2 PidTagNativeBody Property

...

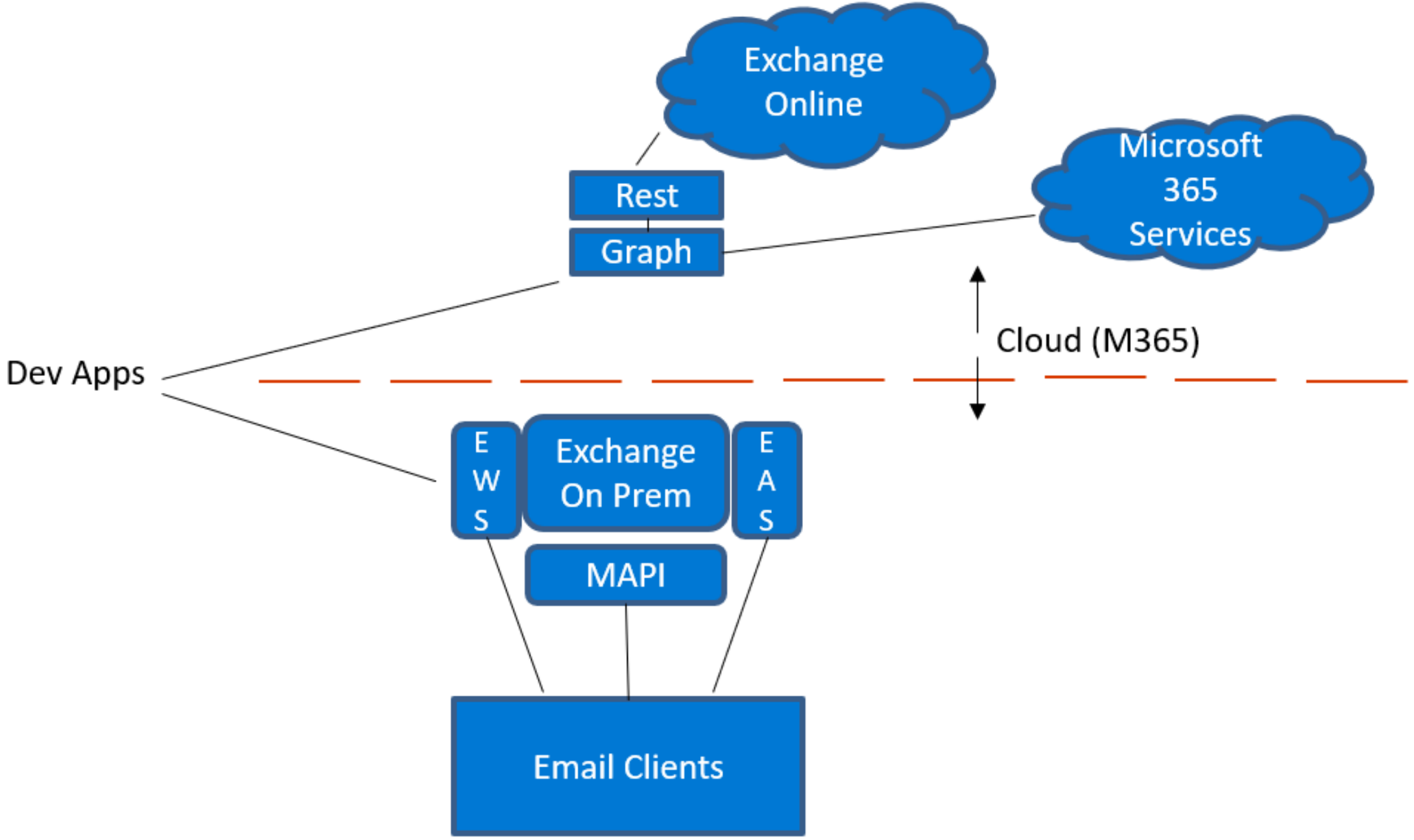
The **PidTagNativeBody** property ([MS-OXPROPS] section 2.805) indicates the best available format for storing the message body<6>

6 Appendix A: Product Behavior

...

<6> Section 2.2.1.56.2: Exchange 2003 and Exchange 2007 do not support the **PidTagNativeBody** property.

EXCHANGE PROTOCOLS: WHAT"S IN SCOPE

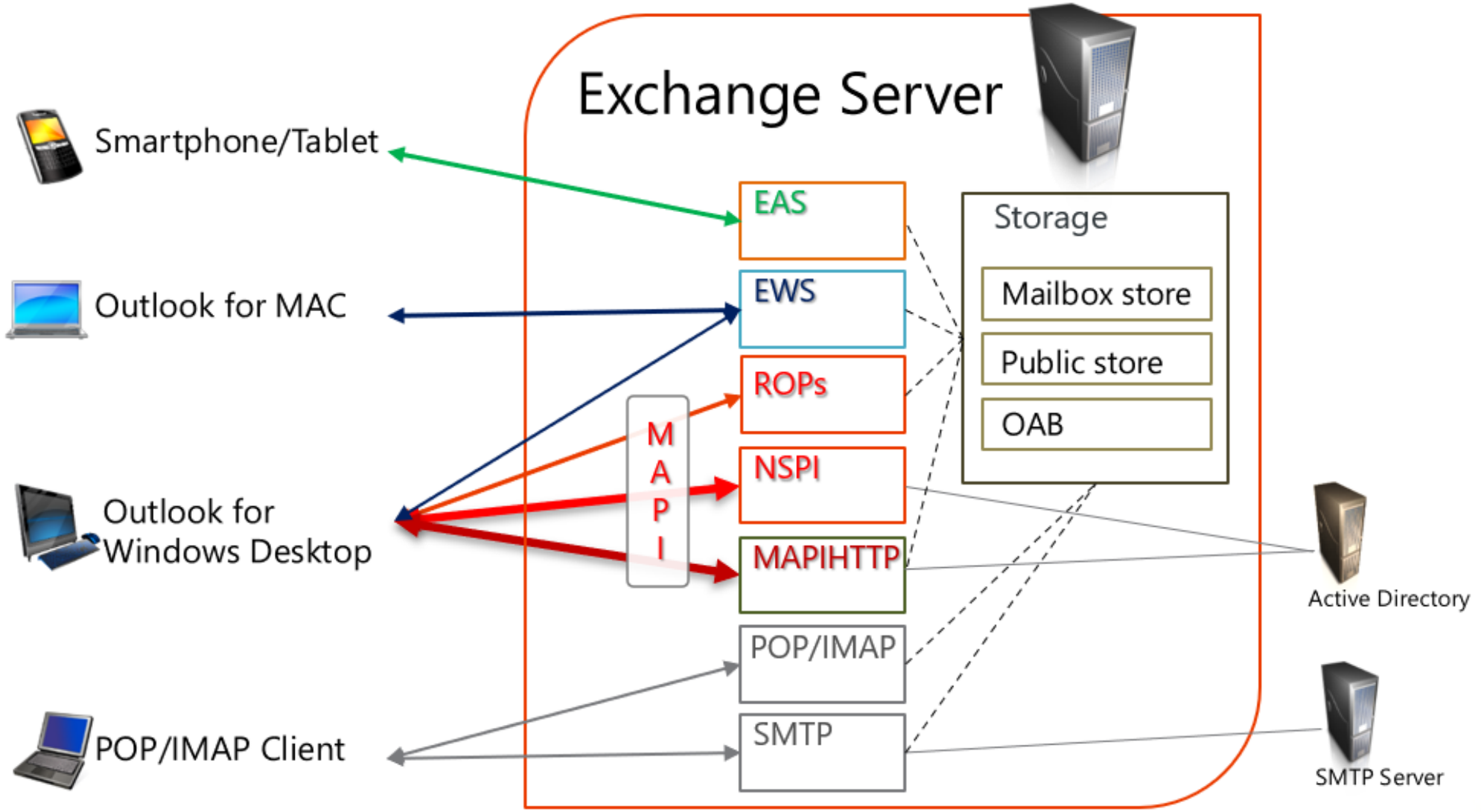

Exchange Online
Microsoft 365 Services
Rest
Graph
Cloud (M365)
Dev Apps
E W S
Exchange On Prem
E A S
MAPI
Email Clients

EXCHANGE PROTOCOLS: DETAILS

Glossary

- **RPC** – Remote Procedure Call
- **ROPs** – Remote Operations
- **MAPI** – Mail API, since 1990th. Originally library used by Outlook for Windows Desktop. Protocol family: **RPC/ROPs**
- **MAPIHTTP** – ROPs over HTTP instead of RPC
- **EWS** – Exchange Web Services (protocol family)
- **EAS** – Exchange Active Sync (protocol family)

Client communication with Exchange Server



There is more to Exchange than email

Calendars	Recurring meetings, cross time zone scheduling
	Availability – both attendees and conference rooms
Tasks	One-time or recurring
	Tracking Due Date and Assignment
Contacts	Multiple Contacts Folders, Suggested Contacts
	Address book for mobile devices
Reminders	For Calendars, Tasks, Emails
Notifications	Server notifies Client that mailbox changed (new mail, etc.)
	Push or Pull

There is more to Exchange than email

Rules	Server Side and Client Side
	Applied on messages on arrival, even when client is not connected
Out of Office	Internal and External recipients can get different OOF messages
	Time restrictions
Mail Tips	Show user that recipient is Out of Office
	Warning that mail includes very large DL, etc.
Compliance	Legal Search (eDiscovery)
	Legal Hold
Archiving	Separate Archive for old emails, accessible by client
	Retention policies set per folder

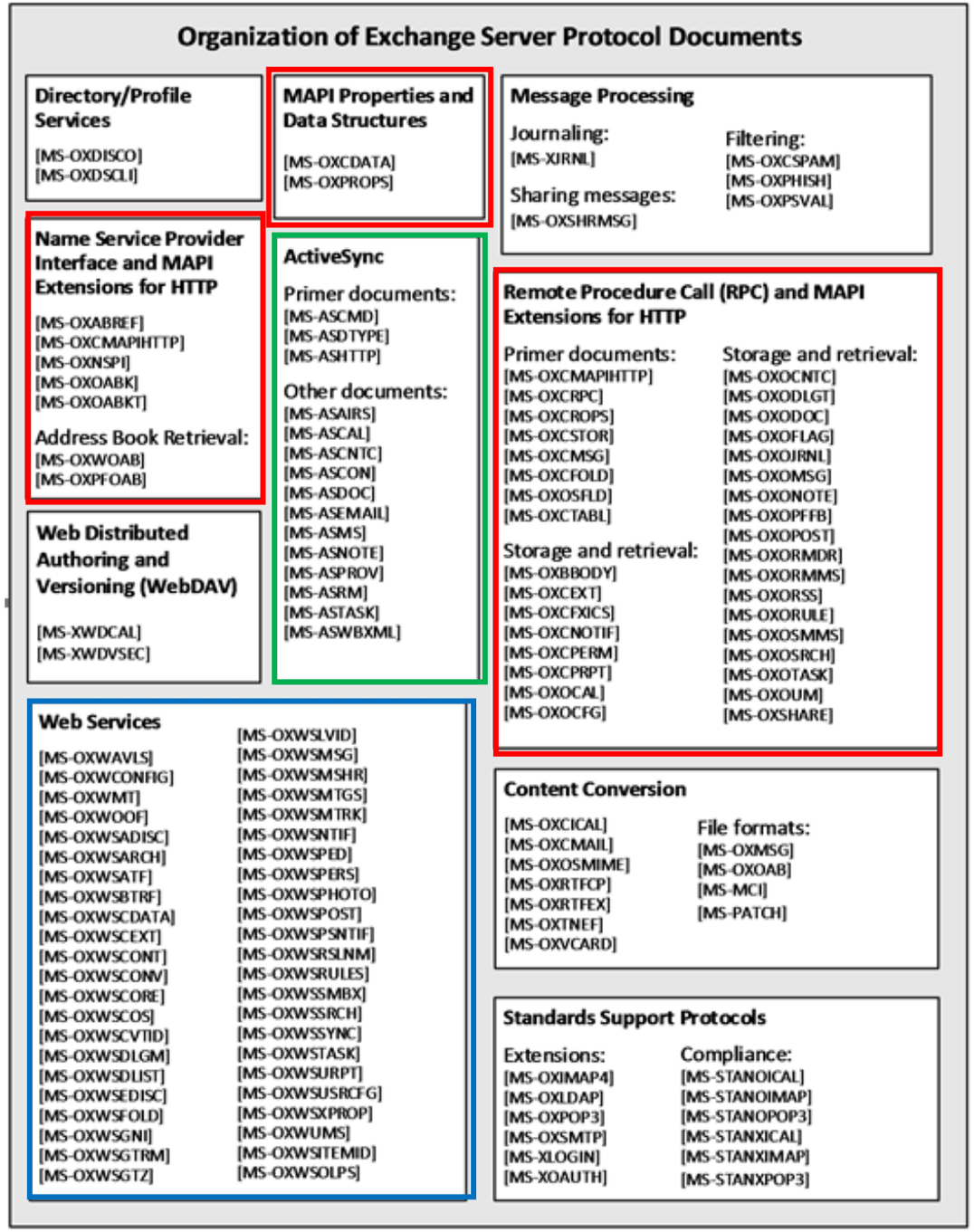
And so on...

Overview Document: MS-OXPROTO

- Defines protocol families
- Scenario-Based
  - How protocols work together
- Walkthroughs and examples
  - Display an e-mail
  - Send an attachment
  - Create an appointment
  - … and other scenarios

Protocol Families

- ROP Protocols
  - Originally RPC transport
  - HTTPS transport introduced in Exchange 2013
  - NSPI is used for Address Book access
- Exchange Web Services (EWS)
- ActiveSync (EAS)
- Other protocol groups
  - Autodiscover
  - Content Conversion
  - Standards Support



EXCHANGE PROTOCOLS: ROP PROTOCOLS

History of terminology – what is "ROP"?

- Initial name: RPC protocols
  - Remote Operations (ROPs) over RPC connection
  - RPC is just a channel to exchange binary blobs between client and server
  - This binary blob can be transmitted by other underlying protocols
    - RPC over TCP or RPC over HTTPS (MS-OXCRPC)
    - Pure HTTPS (MS-OXCMAPIHTTP)
- MAPIHTTP is RPC replacement (Exchange Server 2013)
  - The same binary blob is now transmitted over HTTP, bypassing RPC
- Since it's not RPC anymore, we now call them ROP Protocols
  - In many places it is still called "Exchange RPC protocols", sometimes MAPI

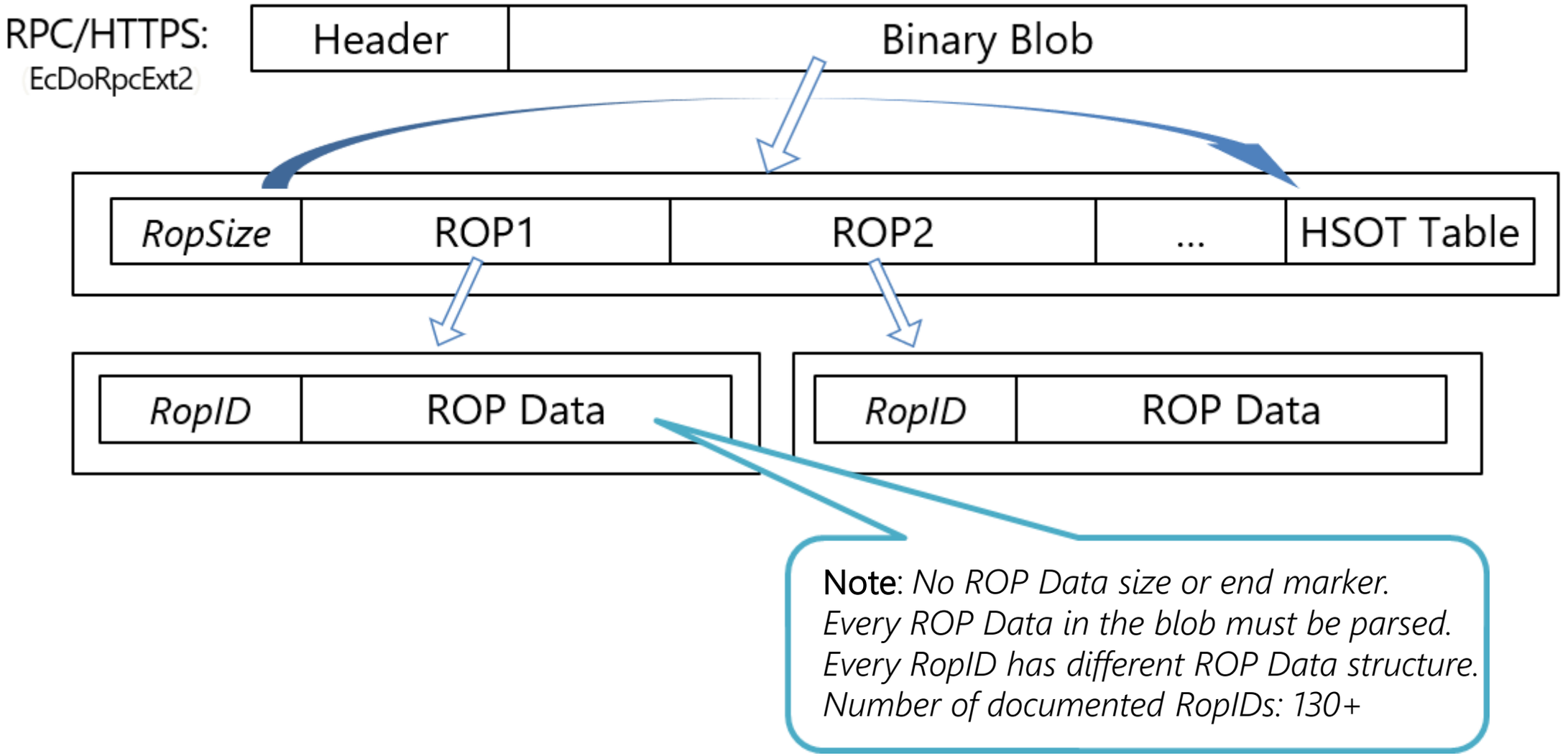
ROP Protocols Details

- Transport layers: RPC/TCP, RPC/HTTPS or pure HTTPS
- Low-level access to Exchange Server Storage
  - Optimized to minimize traffic on the wire
  - Very complex parsing
- Used by MAPI
  - Major channel for Outlook for Windows Desktop communication with Exchange Server
- Originally implemented in Exchange 4.0 (first release of Exchange)
  - Extended and re-architected several times

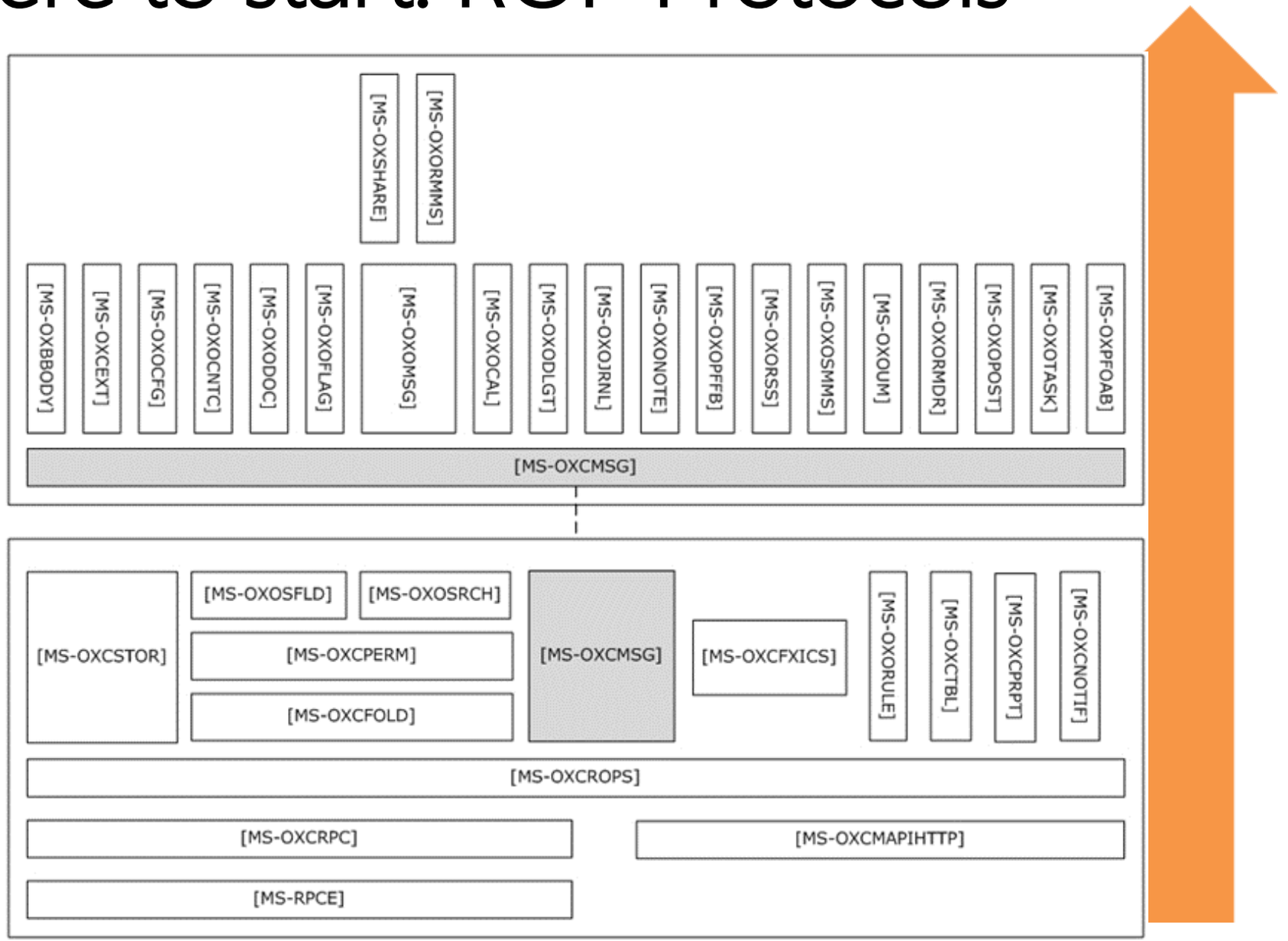
ROP Protocols Documentation

- ROPs protocols define both Server and Client behaviors
- Client behaviors are very complex
  - Wrapped in MAPI implementation on Outlook for Windows Desktop
  - Client performs logic to maintain complex items: Messages, Folders, Calendar, Contacts, Tasks, etc…
- Document Naming: [MS-OXO*] and [MS-OXC*]
- Over 40 Protocols, ~3000 pages

Example of complexity in ROP Protocols



Where to start: ROP Protocols



ROP Protocols Summary

- Not recommended for new applications due to complexity
- Modern Exchange Server has good alternatives
  - Exchange Web Services for On-Prem Server
  - Microsoft Graph for Exchange Online

EXCHANGE PROTOCOLS: EXCHANGE WEB SERVICES (EWS)

EWS Protocols: Details

- Alternative to ROP protocols
- Higher level of abstraction compared with ROP protocols
  - EWS implements messages, attachments, calendar events, contacts on server side
  - Outlook for Windows Desktop uses subset of EWS protocols
    - Unified Messaging, MailTips, Availability, OOF, Office Apps, Room List, Archive, Mailbox Policies, Calendar Sharing, Site Mailboxes, etc.
  - Intuitive/readable XML (easier troubleshooting)
- Transport: SOAP over HTTP(S)

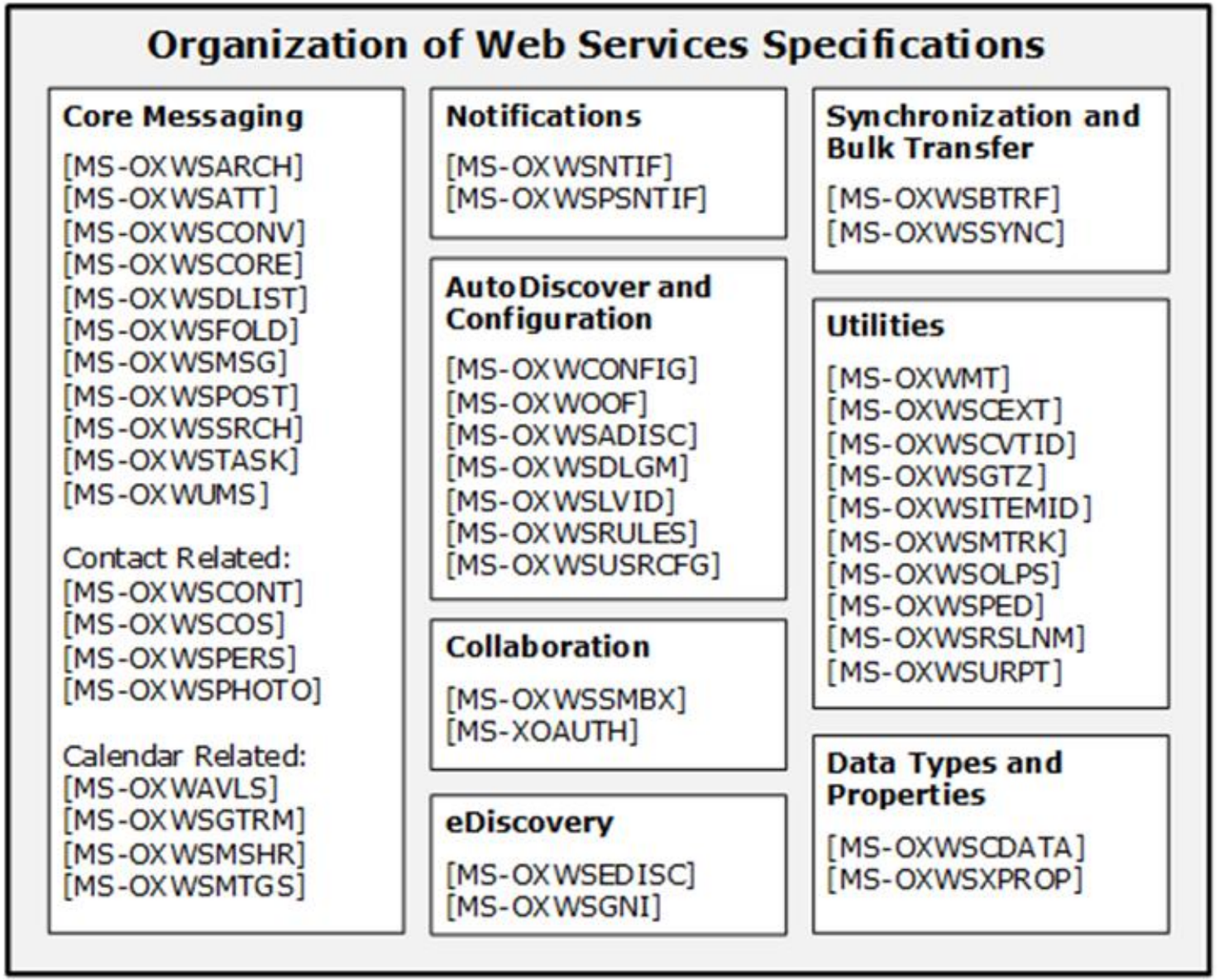
EWS Protocols: Documentation

- Code in any language/platform that supports HTTP/SOAP calls
  - SDKs are available as well
- Document Naming: [MS-OXW*]
  - 43 Protocols, ~2000 pages
- First implementation: Exchange Server 2007
- New on-prem features tend to be implemented in EWS

Where to start: EWS

- Not hierarchical
- Start with

  MS-OXWSCORE



EWS SDKs

- Exchange Web Services (EWS) Managed API
  - https://www.nuget.org/packages/Microsoft.Exchange.WebServices
  - https://github.com/OfficeDev/ews-managed-api
  - https://learn.microsoft.com/en-us/exchange/client-developer/exchange-web-services/explore-the-ews-managed-api-ews-and-web-services-in-exchange

EXCHANGE PROTOCOLS: EXCHANGE ACTIVE SYNC (EAS)

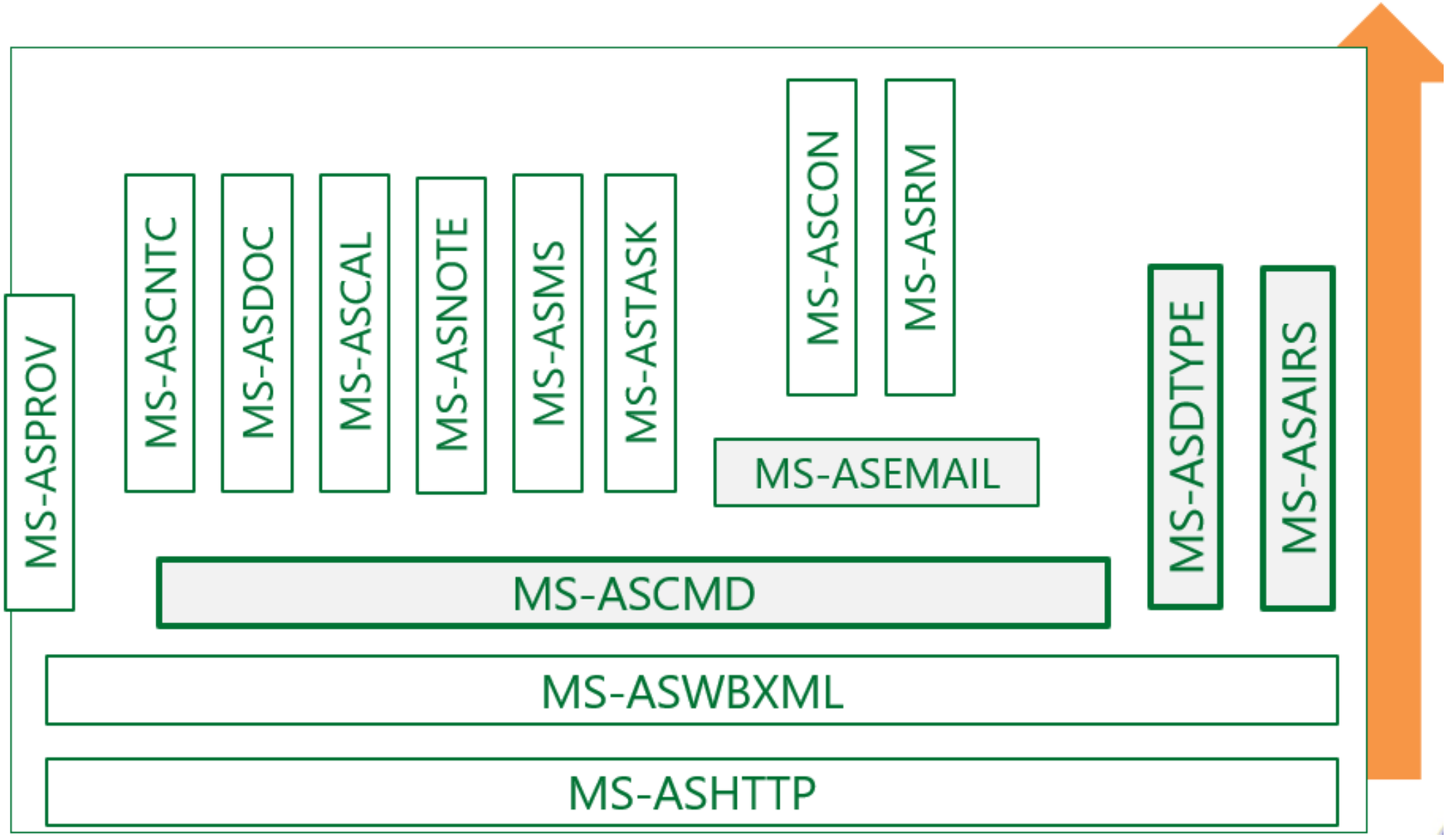
Exchange ActiveSync: Protocols Details

- Lightweight synchronization protocol for Exchange Server
  - Optimized to work on high-latency and low-bandwidth networks
  - Designed to minimize device power usage
- High level of abstraction, similar to EWS in complexity
  - Provides access to email, calendar, contacts, tasks, documents, etc…
- Transport Layer: WBXML over HTTPS
- Industry standard
  - Several non-Exchange Server implementations
  - Wide range of clients – iOS, Android, Windows, etc.

Exchange ActiveSync: Protocols Documentation

- Document Naming: [MS-AS*]
  - 15 Protocols; ~900 pages
- First Implementation: Exchange Server 2003

Where to start: Exchange ActiveSync



Typical Usage of Exchange ActiveSync

- Mobile Applications
- Tablets and lightweight desktop applications
  - Allows for low-bandwidth and high-latency data (Internet) connections while scaling reasonably on high-speed connections.
- Additional considerations
  - Not feature parity with Exchange ROPs and Web Service (EWS) protocols
  - Licensing requirements

CHOICE OF PROTOCOLS TO USE

On-Premises Choice

- EWS is recommended for new applications
- ROPs protocols are very complex
- EAS has special purpose

Exchange Online choice: Graph APIs

- If you develop for Microsoft 365, consider Microsoft Graph APIs
  - Modern authentication
  - Seamless integration with other services (SharePoint, OneDrive, Azure, etc.)
  - New Exchange Online features are implemented in Microsoft Graph APIs

Microsoft Graph APIs Starting Points

- Microsoft Graph Dev Center | APIs and app development
- https://devblogs.microsoft.com/microsoft365dev/announcing-30-days-of-microsoft-graph-blog-series/
- https://developer.microsoft.com/en-us/graph/graph-explorer

RESOURCES

Exchange Protocol Test Tools

- Protocol Test Suites
  - Protocol families: EAS, EWS, and ROPs (RPC/MAPIHTTP)
- Fiddler Inspectors:
  - Protocol families: ROPs (MAPIHTTP)

Exchange Protocol Resources

- All Exchange protocol documents
  - https://learn.microsoft.com/en-us/openspecs/exchange_server_protocols
- Office Interoperability Blog:
  - https://techcommunity.microsoft.com/t5/exchange-team-blog/bg-p/Exchange
- Protocol Test Suites
  - https://github.com/OfficeDev/Interop-TestSuites
- Fiddler inspectors for Office and Exchange protocols
  - https://github.com/OfficeDev/Office-Inspectors-for-Fiddler
- Help with Open Specifications:
  - mailto:dochelp@microsoft.com

Microsoft

Thank you!