

Decoding IP terminology: Part 1



August 18th 2015 - 10:00 AM

By Gary Olson, GHO Group



Image courtesy pm-partners group

You don't need a "secret decoder ring" to understand IP terminology. However, if you think that a DAM involves water, that SOAP is for the shower, that PAM is your cousin's name and ESB is short for Bruce Springsteen's E-Street Band, read on. You have some catching up to do.

As I was sitting in a meeting on Media Asset Management a younger colleague sitting next to me commented that she didn't understand most of the terms the presenter was using. It's been said that all

business sectors have their own abbreviations and mnemonics AND that people in each sector could easily converse with each using only 3 letters and 3 numbers. This is too much information to put in one article so I will break it into bite size chunks.

The broadcast industry is no different. We have had Ike's, BVW's, and K-Scope's. Now we use ASI, 3G, 10G, MADI, 264 and 265. We have a whole new lexicon and even the name of all this new stuff is an abbreviation or contraction – IP.

IP is the contraction of TCP/IP which is the abbreviation of Transmission Communication Protocol/ Internet Protocol. Quite the mouthful, this was the first communication language for Ethernet Networks and between the networks and servers that made up the public Internet. (More about this in The Anatomy of a Network series)The contraction to IP has been adopted as the generic description for all technologies, systems and devices connected over a network. It's easier to say and remember than TCP/IP. So we had analog, then digital which became SD-SDI, then HD-SDI, then SD/HD SDI now just SDI. So it's only logical that we went from TCP/IP to IP. Even the term protocol has gotten a little over used.

So what triggered the need to explain all this!



A common terminology is key to any segment's communication. Participants in that particular space must understand the shorthand, lest they be judged as less qualified or uninformed.

SOA, SOAP and RESTful to begin with. Of course there's SaaS, XML, SDK, ETL and API's. What about Taxonomy, Ontology, CMS, DAM, MAM and PAM. Then let's not forget PPTP, NTP, QoS and MPLS. Of course there's SDN, NFV, SDS, EBS and SVDN. We have routers, switches, gateways and access points. Software services, stacks, protocols and middleware, OH and let's not forget platforms, CDN's, 4G, LTE, 802.11x and 5G. Have I missed any? Probably!

The road to IP is littered with adopted terminology from the IT and Telecom industry as well as new abbreviations as new technologies are introduced. There's all kinds of terminology for hardware, software and communications.

So it occurred to me, a small guide would help in understanding what each of these mean. More than a Wiki definition. What they actually mean and where they are used.

SOA – Service Oriented Architecture: **SOA** is a type of software architecture. It's not something you buy, it's something you do. It is not a product or application that can be bought or is it programming language. SOA is more of a technique or philosophy for developing software applications. At its core, an SOA architecture is the philosophy used in software development that allows the software services to be reused and make it unnecessary to start each development effort from scratch when it's time to build upgrades and modifications.

SOAP was originally an acronym for **S**imple **O**bject **A**ccess **P**rotocol that is not used anymore. The current version is XML and maintained by the XML Working Group. SOAP is a communication specification for exchanging structured information when implementing web services on a network. It works within the Application Layer 7 of the OSI Model using HTTP and SMTP for messages and transmission. One example of how SOAP is used would be that an Editing application sends a SOAP request to a server that has web services enabled like a MAM system with a set of search criteria. The MAM server sends a SOAP response with the data in XML format e.g. title, location, length and file size. The data generated is in a standardized machine-readable format, the requesting application can integrate it directly without any conversion or translation.

RESTful Representational State Transfer (REST) is used by browsers and can almost be thought of as the language of the Internet. Pretty bold statement. No really. As cloud usage continues to develop there are various application programming interfaces (APIs) emerging to allow servers running Web services that integrate with cloud based services. REST was elected as the programming choice for building APIs that allow applications to connect and interact with cloud services. As an example RESTful APIs are used by many sites, including Google, Amazon, Twitter and LinkedIn.

Stacks – This is term used to describe software layers that make up a whole system. TCP/IP the combination is a stack. Automation sitting over a playout application is a stack. In the cloud, stacks are used to describe the architecture how to interface with cloud solutions.

SaaS – Software as a Service – one of those Cloud things.– An example of this is Encoder.com, accessible by browser, cloud based, and a software application living in the cloud providing a service over the Internet.

API – Application Protocol Interface – is the programming code that is used to develop the interface to an application. The API provides code for correct syntax and function calls needed by a programmer to develop communications between applications. An example of this is sending metadata from a statics program into the asset manager to add more description to a piece of content.

SDK – Software Development Kit – This is a set of programs provided by the application developer that is used write the API. The SDK is important when working with well-established programs or operating systems. An example of this is using an SDK from Adobe, Autodesk or Microsoft to write a program or interface to their systems. They will not allow a user or outside programmer access to the core application but they will provide tools to create programs that will work and interface with them.



Middleware – this is the IP version of digital glue. These are third party applications used to interface applications. This is very common working with databases to move data between them while respecting the permission and security profile of the database. Extract – Transfer – Load [ETL] is an example of Middleware. ETL is a third party tool for integrating different databases establishing a connection and operates in the background of a primary application

XML – is a programming language and protocol widely accepted as the standard for applications to share information. While the language is standardized the format of the data is not. So data from one application sent as XML will need to be mapped into the appropriate structure in the receiving application.

AND the last entry for Part 1 is **ESB!**

ESB – Empire State Building or Extra Special Bitters – OK How about **Enterprise Service Bus** !ESB has been floating around the IT space for a while. The term "bus" is familiar using a computer analogy where the internal bus connects the CPU, RAM and controllers. Or in a card frame for distribution and processing equipment, the backplane is also considered a bus where the cards plug-in. ESB is an application bus where applications connect and ESB is the "back plane" that provides common control, monitoring and management of connect applications. Similar to the way the control surface of the card frame works.

Whew, this seems like a lot to digest in one sitting, so since this is Part 1, I will pick it up again in Part 2.

Follow my continuing adventure as I attempt to [Smooth the Rocky Road to IP](#). My goal is to explain, debunk and educate. And sometimes just ramble on.

Readers can find more of Gary Olson's tutorial series on IP technology on [The Broadcast Bridge](#).

[The Anatomy of the IP Network, Part 1](#)

[The Anatomy of the IP Network, Part 2](#)

[The Anatomy of the IP Network, Part 3](#)

[The Last Mile Is The Hardest--Getting To Live IP](#)

[NAB Musings on the Move to IP](#)

[Smoothing The Rocky Road to IP](#)

[Live Video Over IP](#)