

## Topiclist

Joe Average

### AAC - Advanced Audio Coding

[AAC](#) is one of two audio coding methods defined by the MPEG-2 standard. It is not backward compatible with MPEG-1 audio coding, which is why it is sometimes also referred to as the "non backward compatible" audio coding of MPEG-2. [AAC](#) provides more sophisticated audio coding than MPEG-2 [BC](#) audio coding (the other audio coding method of MPEG-2, better known as MP3), but it is not as widely spread.

### ABNF - Augmented Backus-Naur Form

Internet technical specifications often need to define a format syntax and are free to employ whatever notation their authors deem useful. Over the years, a modified version of [BNF](#), called [ABNF](#), has been popular among many [Internet](#) specifications. It balances compactness and simplicity with reasonable representational power.

### ACE - ASCII Compatible Encoding

An [ACE](#) a string of characters resulting from a particular algorithm for transforming multilingual character information into an [ASCII](#)-based alphanumeric form acceptable by the existing [DNS](#). This means that an [ACE](#) encoded string must conform to the [LDH](#) restrictions for strings.

### Achievement

The [Achievement](#) is a base topic for topics describing a general technical achievement.

### ActiveX

[ActiveX](#) is the name Microsoft has given to a set of object-oriented concepts, technologies, and tools. It is important to notice that [ActiveX](#) is not a specific technology, but a brand name, what it's applied to can vary over time. However, the main technology is [COM](#). Used in a network with a directory and additional support, [COM](#) becomes the [DCOM](#). The main object that is created when writing a program to run in the [ActiveX](#) environment is a component, a self-sufficient program that can be run anywhere in the [ActiveX](#) network. This component is known as an "[ActiveX](#) control". An [ActiveX](#) control is roughly equivalent to a [applet](#). One important difference is that an [ActiveX](#) control must be compiled for a specific platform, while an [applet](#) is platform-independent.

### AES - Advanced Encryption Standard

[AES](#) is the replacement of [DES](#), because recent research and attacks have shown that DES is not as safe as necessary for some applications. [AES](#) has been chosen from a number of candid-

ates based on criteria such as efficiency, robustness, and cryptographical safety.

## AIFF - Audio Interchange File Format

[AIFF](#) is a proprietary [Audio Format](#) developed by Apple. The format can store monaural or multichannel sampled sounds in a range of sample rates and sample resolutions. Although originally [AIFF](#) did not support compressed audio data, a new version of the format called [AIFF-C](#) has been defined which allows compression.

## AIFF-C - AIFF Compressed

[AIFF-C](#) is an extended version of Apple's [AIFF Audio Format](#). It incorporates compression features, which have not been included in the original [AIFF Audio Format](#).

## ANSI - American National Standards Institute

[ANSI](#), founded in 1918, does not itself develop American National Standards; rather it facilitates development by establishing consensus among qualified groups. The Institute ensures that its guiding principles (consensus, due process, and openness) are followed by the more than 175 distinct entities currently accredited. [ANSI](#) promotes the use of US standards internationally, advocates US policy and technical positions in international and regional standards organizations, and encourages the adoption of international standards as national standards where these meet the needs of the user community.

## Apache Software Foundation

The [Apache Software Foundation](#) is a collaborative software development effort aimed at creating a robust, commercial-grade, feature-full, and freely-available source code implementation of an [HTTP Server](#). The project is jointly managed by a group of volunteers located around the world, using the [Internet](#) and the [WWW](#) to communicate, plan, and develop the [Server](#) and its related documentation. In addition, hundreds of users have contributed ideas, code, and documentation to the project. In addition to the well-known and established Apache HTTP Server, the [Cocoon](#) project aims at developing a next-generation [HTTP Server](#) integrating [XML](#) technologies.

## API - Application Programming Interface

An [API](#) is an [Interface](#) which is used for accessing an application or a service from a program. An [API](#) makes it possible to use programs from within programs, therefore it is the foundation for modular systems with clearly defined [Interfaces](#) between separate components. In a way, an [API](#) can be regarded as the local equivalent of a [Protocol](#), because it is used for the same purposes and defines the same things (the possible interactions between components, and the data that is exchanged while interacting). However, traditionally [APIs](#) are used for [Interfaces](#) on one computer, while [Protocols](#) are used for distributed scenarios.

## applet

An [applet](#) is a special type of [Java](#) program that can be included in an [HTML](#) page, much as

an image can be included. When using a [Java-compatible Browser](#) to view a page that contains an [applet](#), the [applet](#)'s code is transferred to and executed by the [Browser](#). Since an [applet](#) is platform-independent, the same [applet](#) can be executed on all types of platforms, as long as they support [Java](#).

### ARPANET

The [ARPANET](#) is the precursor to the [Internet](#). It was established through an ARPA-funded research program in 1968 and provided the foundation of the [Internet](#) by providing the [Protocols](#) as well as the infrastructure. The original [ARPANET](#) grew into the [Internet](#).

### ASCII - American National Standard Code for Information Interchange

[ASCII](#) specifies the coding of space and a set of 94 characters (letters, digits and punctuation or mathematical symbols) suitable for the interchange of english language documents. [ASCII](#) forms the basis for most computer code sets and is the american national version of ISO 646.

### ASF - Advanced Streaming Format

Microsoft's [ASF](#) is an extensible file format designed to store synchronized multimedia data. It supports data delivery over a wide variety of networks and protocols while still proving suitable for local playback. The explicit goal of [ASF](#) is to provide a basis for industry-wide multimedia interoperability. Each [ASF](#) file is composed of one or more media streams. The file header specifies the properties of the entire file, along with stream-specific properties. Multimedia data, stored after the file header, references a particular media stream number to indicate its type and purpose. The delivery and presentation of all media stream data is synchronized to a common time-line. [ASF](#)'s functionality is similar to the one provided by Apple's [QuickTime](#).

### ASN.1 - Abstract Syntax Notation One

[ASN.1](#) defines a method for the specification and encoding of arbitrary data structures. [ASN.1](#) is part of the [OSI](#) model of protocol layers, it is located within the "presentation layer" (layer 6). The approach of [ASN.1](#) is to define an abstract syntax for the specification of structured data, and encoding rules for transforming structured data into a binary representation (which can then be exchanged). BER is the oldest encoding for [ASN.1](#), and a subset of it (known as DER) is used for [X.509 Certificates](#).

### ASP - Application Service Provider

An [ASP](#) is providing an application to be used by [ASP](#) customers. In contrast to traditional software distribution, the software that provides the service is run by the [ASP](#), and it is accessed remotely (and thus never downloaded or installed) by the customers. A common way for providing such services is over the [WWW](#), where the application is accessed via various [HTML](#) pages which in their entirety make up the [GUI](#). One of the best known [ASPs](#) is Hot-Mail, which makes an Email application available over the [WWW](#).

### Audio Compression

The [Audio Compression](#) is a base topic for topics describing a defined way of compressing audio.

### Audio Format

The [Audio Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer of audio.

### AVI - Audio Video Interleave

Microsoft's [AVI](#) file format is used for storing audio and/or video information. It is a common format for audio and video files within PC environments. Being a proprietary technology, [AVI](#) can be functionally compared to Apple's [QuickTime](#).

### B2B - Business-to-Business

[B2B](#) is the use of computer applications communicating over networks to allow businesses to complete a transaction or part of a transaction. In contrast to [B2C](#), [B2B](#) describes the interaction of businesses. While the amount of [B2C](#) transactions using [WWW](#)-based technologies (often called online shopping) exceeds the amount [B2B](#) transactions, the total value of the [B2B](#) transactions is much larger than that of the [B2C](#) transactions, because typically [B2B](#) transactions are of much greater value than [B2C](#) transactions.

### Base64

[Base64](#) encoding is designed to represent arbitrary sequences of octets in a form that need not be humanly readable. The encoding and decoding algorithms are simple, but the encoded data are consistently only about 33 percent larger than the unencoded data. In [Base64](#), a 65-character subset of [ASCII](#) is used, enabling 6 bits to be represented per printable character. [Base64](#) is virtually identical to the encoding used in PEM applications.

### BC - Backward Compatible

[BC](#) audio coding is one of two audio coding methods defined by the MPEG-2 standard. It is backward compatible with MPEG-1 audio coding. [BC](#) is comprised of three different and increasingly complex layers, and if all three layers are used, it is the audio coding most commonly referred to as MP3. MPEG-2 also defines a non backward compatible audio coding, which is known as [AAC](#).

### BibTeX

[BibTeX](#) is a simple but popular [Format](#) and program for storing and processing bibliographic references. In almost all cases it is used together with [LaTeX](#), but it can also be used to create other kinds of output.

### BibTeXML - BibTeX Markup Language

[BibTeXML](#) is an [XML](#) representation for [BibTeX](#) data. The advantage of [BibTeXML](#) over

the standard [BibTeX](#) format is the availability of tools for processing [XML](#) data in general, and the ease of integration of [BibTeXML](#) into an [XML](#)-based environment.

### BIND - Berkeley Internet Name Domain

[BIND](#) implements an [Internet](#) name server. It consists of a server and a resolver library. [BIND](#) is an implementation of [DNS](#), both [Server](#) and [Client](#). Development of [BIND](#) is funded by the ISC. [BIND](#) has been ported to Windows and VMS, but is most often found on Unix. [BIND](#) source code is freely available and very complex; most of the development on the [DNS Protocols](#) is based on this code; and most Unix vendors ship [BIND](#)-derived [DNS](#) implementations. As a result, the [BIND](#) name server is the most widely used name server on the [Internet](#).

### BNF - Backus-Naur Form

[BNF](#) is a formal meta-syntax used to express context-free grammars. [BNF](#) is one of the most commonly used meta-syntactic notations for specifying the syntax of programming languages, command sets, PDUs, and similar things. However, pure [BNF](#) is rather limited, so the two variations [EBNF](#) and [ABNF](#) have become more popular.

### Body

The [Body](#) is a base topic for topics describing a single person or a body consisting of several persons.

### Bridge

A [Bridge](#) is a device that connects network segments on the data link layer of the network. Related to [Bridges](#) are [Repeaters](#), [Routers](#), and [Gateways](#), which also connect network segments, but on different layers of the networking architecture.

### Browser

The [Browser](#) is a base topic for topics describing programs which are a [HTTP Client](#) and provide a UI for browsing through the [WWW](#). A [Browser](#) communicates with an [HTTP Server](#).

### CA - Certification Authority

A [CA](#) is an authority trusted by one or more users to create and assign [Certificates](#). Optionally the [CA](#) may create the user's keys. It is important to note that the [CA](#) is responsible for the [Certificates](#) during their whole lifetime, not just for issuing them.

### Canonical XML

[Canonical XML](#) refers to [XML](#) that is in canonical form. XML canonicalization is a process that generates the [Canonical XML](#) of a given [XML](#) document. [XML](#) specifies the syntax of [XML](#) documents. [XML Namespaces](#) specifies additional syntax and semantics for [XML](#) documents. Because of the syntax of [XML](#), it is possible for [XML](#) documents which are equivalent

ent for the purposes of many applications to differ in physical representation. For example, they may differ in whitespace occurrences, their entity structure, attribute ordering, and character encoding. [Canonical XML](#) establishes a method for determining whether two documents are identical, or whether an application has not changed a document, except for transformations permitted by [XML](#) and [XML Namespaces](#).

## CCITT - Consultative Committee on International Telegraphy and Telephony

Former name of the [ITU](#) before it was renamed in 1993.

## ccTLD - Country-Code TLD

A [ccTLD](#) is a [DNS TLD](#) identifying domain names for a given country. The country-codes being used are the two letter codes defined by ISO 3166. The IANA is not in the business of deciding what is and what is not a country. The selection of the ISO 3166 list as a basis for [ccTLDs](#) was made with the knowledge that ISO has a procedure for determining which entities should be and should not be on that list.

## Certificate

[Certificates](#) are digital documents attesting to the binding of a public key to an individual or other entity. They allow verification of the claim that a given public key does in fact belong to a given individual. [Certificates](#) help prevent someone from using a phony key to impersonate someone else. In their simplest form, [Certificates](#) contain a public key and a name. As commonly used, a [Certificate](#) also contains an expiration date, the name of the [CA](#) that issued the [Certificate](#), a serial number, and perhaps other information. Most importantly, it contains the digital signature of the certificate issuer. The most widely accepted format for certificates is [X.509](#), thus, [Certificates](#) can be read or written by any application complying with [X.509](#).

## CGA - Color Graphics Adapter

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 320 by 200 pixels.

## CGI - Common Gateway Interface

[CGI](#) is an [API](#) for interfacing external applications with information [Servers](#), such as an [HTTP Server](#). [CGI](#) can be used with any [Programming Language](#). A plain [HTML](#) document that the [HTTP Server](#) retrieves upon a request is static, which means it exists in a constant state, for example as a text file that does not change. A [CGI](#) program, on the other hand, is executed in real-time, so that it can generate dynamic information.

## CGM - Computer Graphics Metafile

[CGM](#) is a machine and [Operating System](#) independent interchange format that provides elements to represent geometric graphics (eg, polygons or circles) and raster graphics (eg, pixel arrays). It consists of a functional specification and multiple encodings for different purposes. There are three standardized encodings for [CGM](#), clear-text, character and binary. Clear-text

is human-readable. Character encoding is more compact, but still uses [ASCII](#) characters, so that it can be interchanged without [Protocol](#) problems. The binary encoding is more compact still, and quick to encode and decode, but completely unreadable.

### Chunked Encoding

With the introduction of persistent connections in [HTTP/1.1](#), the length of a resource which is sent in a response can no longer be implicitly signaled by closing the connection. However, for the majority of resources, the length is known in advance and can be given in the [HTTP's](#) Content-Length header field. For all other resources (such as dynamically created content), [Chunked Encoding](#) can be used. [Chunked Encoding](#) transfers the message body as a sequence of chunks of known length.

### CLF - Common Log Format

[CLF](#) is a log file [Format](#) for [HTTP Servers](#), containing information about the host, identification of the user (if available), authorized user-name (if available), date, the request line itself, and the returned status and number of bytes. Although most [HTTP Servers](#) can be configured to produce other [Formats](#) of log files, many tools exist to analyze [CLF](#) files, so custom log file [Formats](#) should only be used if absolutely necessary.

### Client

The [Client](#) is a base topic for topics describing clients. A [Client](#) is one component in the [Client/Server-Model](#) and contacts a [Server](#) for requesting a service of some kind.

### Client Side Technology

The [Client Side Technology](#) is a base topic for topics describing a [Technology](#) which is used on the [Client](#) side of a scenario based on the [Client/Server-Model](#).

### Client/Server-Model

The [Client/Server-Model](#) is a base topic for topics describing models of interacting [Clients](#) and [Servers](#). In the [Client/Server-Model](#), a [Server](#) waits for requests from [Clients](#), and after receiving such a request, the [Server](#) processes it and send a response. The communication between the two peers is based on a [Protocol](#), which is defining the possible interaction patterns and the information being exchanged.

### CMS - Cryptographic Message Syntax

[CMS](#) is used to digitally sign or encrypt arbitrary messages. [CMS](#) describes an encapsulation syntax for data protection. It supports digital signatures and encryption. The syntax allows multiple encapsulation, so one encapsulation envelope can be nested inside another. Likewise, one party can digitally sign some previously encapsulated data. It also allows arbitrary attributes, such as signing time, to be authenticated along with the message content, and provides for other attributes such as counter-signatures to be associated with a signature.

### Cocoon

**Cocoon** is an **XML** publishing framework based on **XSLT**. Designed for performance and scalability around pipelined **SAX** processing, **Cocoon** offers a flexible environment based on the separation of content, logic, and style. **Cocoon**'s centralized configuration system and caching are designed to create, deploy, and maintain scalable **XML** applications. **Cocoon** interacts with most data sources, including **File Systems**, **RDBMSs**, **LDAP**, **XML Database Management Systems**, and network-based data sources. It adapts content delivery to the capabilities of different output formats such as **HTML**, **WML**, **PDF**, **SVG**, **RTF**, and others. **Cocoon** can be executed as a servlet as well as through a command line interface.

### Codec - Coder/Decoder

A **Codec** is an entity that is responsible for encoding and decoding some **Format**. **Codecs** can be implemented in hardware or software. A typical example is an **MP3 Codec**, which is responsible for decoding an **MP3** audio stream for playback, or encoding it for transmission or storage using the compact **MP3 Audio Compression**.

### Color Space

The **Color Space** is a base topic for topics describing ways for representing color.

### COM - Component Object Model

Microsoft's **COM** is a software architecture that allows applications to be built from binary software components. **COM** is the underlying architecture that forms the foundation for higher-level software services, like those provided by **OLE**. **COM** defines a binary standard for function calling between components, a way for components to dynamically discover the interfaces implemented by other components, and a mechanism to identify components and their interfaces uniquely.

### Company

The **Company** is a base topic for topics describing a commercial Organization.

### Component Model

The **Component Model** is a base topic for topics describing component models. A **Component Model** is a framework for creating usually complex software from small, re-usable software components.

### Compression

The **Compression** is a base topic for topics describing a **Format** for compressed data.

### Concept

The **Concept** is a base topic for topics describing an abstract concept.

## Consortium

The [Consortium](#) is a base topic for topics describing an organization consisting of other organizations, usually companies.

## Content Negotiation

[Content Negotiation](#) is an [HTTP](#) mechanism which is used to make a selection between different representations for a resource. Different representations can be characterized by language, quality, encoding, or other parameters which do not affect the content of a resource. [HTTP](#) defines two types of [Content Negotiation](#), [Server-driven](#) and [agent-driven](#). In [Server-driven Content Negotiation](#), the [HTTP Server](#) makes the selection and sends a response with the representation of a requested resource which it thinks matches the user's needs, based on the request, available representations, and other information. In [agent-driven Content Negotiation](#), the [HTTP Server](#) responds with a list of all representations and the [HTTP Client](#) (or the user) makes the selection and requests the selected representation.

## Cookie

Originally introduced by Netscape Communications, [Cookies](#) are a general mechanism which [HTTP Server](#) side applications, such as [CGI](#) scripts, can use to both store and retrieve information on the [HTTP Client](#) side of the connection. Basically, [Cookies](#) can be used to compensate for the stateless nature of [HTTP](#). The addition of a simple, persistent, client-side state significantly extends the capabilities of [WWW](#)-based applications.

## CORBA - Common Object Request Broker Architecture

[CORBA](#) describes the architecture of a middleware platform that supports the implementation of applications in distributed and heterogeneous environments. The [CORBA](#) standard is issued by [OMG](#). In contrast to other middleware platforms such as Microsoft's [DCOM](#), [CORBA](#) is a specification that does not prescribe any specific technology.

## CRVX - Character Repertoire Validation for XML

[XML Schema Languages](#) cover different areas of [XML](#) schema aspects, such as grammar-based schemas (e.g., [DTD](#) and [XML Schema](#)) and rule-based schemas (e.g., [Schematron](#)). [CRVX](#) is a specialized and simple schema language for specifying character repertoire constraints for [XML](#) documents. It is meant as a complement for other schema languages which are often geared towards structural constraints for [XML](#) documents. [CRVX](#) is based on the [Unicode](#) character set which is the foundation of [XML](#).

## CSS - Cascading Style Sheets

[CSS](#) is a [Style Sheet Language](#) which has been primarily designed for [HTML](#), even though it can also be used for [XML](#) documents. While [HTML](#) should be used to define the contents of a [WWW](#) page, [CSS](#) is the language for specifying the presentation aspects of it. The two main advantages of [HTML](#) with [CSS](#) over [HTML](#) without [CSS](#) are the clear separation of content and presentation (which makes the automated processing of [WWW](#) pages much easier and

also enables users to apply their own style sheets instead of the defaults provided by their [Browser](#) or a [WWW](#) page's designer), and the greatly enhanced formatting capabilities with [CSS](#).

### CSS-OM - CSS Object Model

[CSS-OM](#) is a model of how [CSS](#) style sheets can be accessed and manipulated through a [DOM](#) interface. [CSS-OM](#) thus is relevant for both [CSS](#) and [DOM](#), and is important if [CSS](#) should be accessed and manipulated from within an application.

### CSS-P - Cascading Style Sheets Positioning

In the first implementation of [CSS](#) in Netscape Communications's Navigator, additional functionality was included which added absolute positioning and layering to the initial CSS1 specification. Netscape Communications submitted a working draft as a proposal for these positioning features to [W3C](#).

### DAP - Directory Access Protocol

[DAP](#) is the original protocol for accessing [X.500](#). Since [DAP](#) is based on [OSI](#) and rather complex, a simplified variant of [DAP](#) based on the [TCP](#) has been designed, which is called [LDAP](#). Today this simpler variant is more popular than [DAP](#) itself.

### DARPA - Defense Advanced Research Projects Agency

[DARPA](#), until 1973 known as ARPA, is the main source for research funds in the United States. In particular, the [Internet](#) (its first infrastructure as well as the protocol suite) originated from the [ARPANET](#), a network which was first designed for US military purposes with the primary design goal to be robust.

### DCE - Distributed Computing Environment

[DCE](#) is an industry-standard, vendor-neutral set of distributed computing technologies. It provides security services to protect and control access to data, name services that make it easy to find distributed resources, and a highly scalable model for organizing widely scattered users, services, and data. [DCE](#) runs on all major computing platforms and is designed to support distributed applications in heterogeneous hardware and software environments.

### DCOM - Distributed Component Object Model

[DCOM](#) is a [Protocol](#) that enables software components to communicate directly over a network in a reliable, secure, and efficient manner. Previously called [OLE](#), [DCOM](#) is designed for use across multiple network transports, including [Internet Protocols](#) such as [HTTP](#). [DCOM](#) is based on DCE RPC and [COM](#).

### DES - Data Encryption Standard

[DES](#) was originally developed at IBM. [DES](#) has been extensively studied since its publication

and is the most well-known and widely used cryptosystem in the world. [DES](#) is an algorithm implementing [Secret-Key Cryptography](#), when used for communications, both sender and receiver must know the same secret key, which is used both to encrypt and decrypt the message. [DES](#) can also be used for single-user encryption, such as to store files on a hard disk in encrypted form. In a multi-user environment, secure key distribution may be difficult. [Public-Key Cryptography](#) provides an ideal solution to this problem. Even though [DES](#) has been cryptographically enhanced by using three encryption rounds, yielding 3DES, the general consensus is that [DES](#) encryption is too weak for some scenarios, and the stronger [AES](#) will eventually replace [DES](#).

### DHTML - Dynamic HTML

[DHTML](#) does not refer to a specific version or a specific feature of [HTML](#). It is an expression which is commonly used to refer to all features of [HTML](#) which go beyond the presentation of static documents. The most popular mechanisms which are encompassed by the term [DHTML](#) are [CSS](#), scripts (in most cases [ECMAScript](#)-based languages, embedded with the `<SCRIPT>` element), and objects (embedded with the `<OBJECT>` element). [DHTML](#) also often refers to [Browser](#)-specific extensions of particular mechanisms, such as extensions to the basic scripting methods, or the ability to dynamically download fonts. The "glue" between the different components which make up [DHTML](#) (mainly [HTML](#), [CSS](#), and a scripting language) is provided by [DOM](#).

### Diffie-Hellman

[Diffie-Hellman](#) key agreement describes a method whereby two parties, without any prior arrangements, can agree upon a secret key that is known only to them (and, in particular, is not known to an eavesdropper listening to the dialogue by which the parties agree on the key). This secret key can be used, for example, to encrypt further communications between the parties using [Secret-Key Cryptography](#). The intended application of this standard is in [Protocols](#) for establishing secure connections. Details on the interpretation of the agreed-upon secret key are outside the scope of [Diffie-Hellman](#) key agreement, as are details on sources of the pseudo-random bits required by this method. The [Diffie-Hellman](#) key exchange is vulnerable to a middleperson attack. This vulnerability is due to the fact that [Diffie-Hellman](#) key exchange does not authenticate the participants.

### Directory

The [Directory](#) is a base topic for topics describing directory systems.

### Display

The [Display](#) is a base topic for topics describing a [Technology](#) for displaying data.

### Display Standard

The [Display Standard](#) is a base topic for topics describing a [Technology](#) for how to display data on a [Display](#).

### Distributed File System

The [Distributed File System](#) is a base topic for topics describing distributed file systems.

### DNS - Domain Name System

[DNS](#) is a distributed, replicated, data query service mainly used on the [Internet](#) for translating host names to IP addresses. The three main components of [DNS](#) are: the domain name space and resource records, which are specifications for a tree structured name space and data associated with the names; name servers, which are server programs which hold information about the domain tree's structure and set information; and resolvers, being programs that extract information from name servers in response to client requests.

### DOI - Digital Object Identifier

The [DOI](#) system is a mechanism for marking digital objects in order to facilitate E-Commerce and enable copyright management in a digital environment. [DOI](#) not only provides a unique identification for digital content, but also a way to link users of the materials to the rights holders themselves to facilitate automated digital commerce. The underlying technology of [DOI](#) is the [Handle System](#), which associates each [DOI](#) name with one or more locations where the object may be found.

### DOM - Document Object Model

[DOM](#) is a platform- and language-neutral [API](#) that allows programs and scripts to dynamically access and update the content, structure and style of [WWW](#) documents (currently, definitions for [HTML](#) and [XML](#) documents are part of the specification). The document can be further processed and the results of that processing can be incorporated back into the presented page. [DOM](#) is a tree-based [API](#) to documents, which requires the whole document to be represented in memory while processing it. A simpler alternative to [DOM](#) is the event-based [SAX](#), which can be used to process very large [XML](#) documents that do not fit into the memory available for processing.

### DSL - Digital Subscriber Line

[DSL](#) is a technology for bringing high-bandwidth information to homes and small businesses over ordinary copper [PSTN](#) lines. The widely used term xDSL refers to different variations of DSL, such as ADSL, HDSL, VDSL and SDSL. Most [DSL](#) technologies require that a signal splitter be installed at a home or business, requiring the expense of a phone company visit and installation. However, it is possible to manage the splitting remotely, this has been standardized as G.Lite.

### DSS - Digital Signature Standard

DSA was published by NIST in the [DSS](#) standard. DSA is for authentication only. In DSA, signature generation is faster than signature verification, whereas in [RSA](#), signature verification is faster than signature generation (if the public and private exponents, respectively, are chosen for this property, which is the usual case). NIST claims that it is an advantage of DSA

that signing is faster, but many people in cryptography think that it is better for verification to be the faster operation.

## DSSSL - Document Style Semantics and Specification Language

**DSSSL** is an international standard for specifying document transformation and formatting in a platform- and vendor-neutral manner. **DSSSL** can be used with any document format for which a property set can be defined according to the HyTime standard. In particular, **DSSSL** can be used to specify the presentation of documents marked up according to **SGML**. **DSSSL** consists of two main components, a transformation language and a style language. The transformation language is used to specify structural transformations on **SGML** source files. The transformation language can also be used to specify the merging of two or more documents, the generation of indexes and tables of contents, and other operations.

### DSSSL-Lite

**DSSSL-Lite** was an early approach to define a profile (ie, a functional subset) of **DSSSL** in an attempt to create a version of **DSSSL** which is less complex and still powerful enough to be sufficient for a large number of applications. **DSSSL-Lite** never became an actual standard, but the work on it was used as input for the **DSSSL-O** activity.

### DSSSL-O - DSSSL-Online

Based on results from the **DSSSL-Lite** activity, **DSSSL-O** was an attempt to define a profile (ie, a functional subset) of **DSSSL**. This profile should be less complex than full **DSSSL** and particularly suited to the needs of online publishing. **DSSSL-O** never became an actual standard, but it was used as the base for **XSL**, which is used as the style sheet language for **XML** documents.

## DTD - Document Type Definition

A **DTD** is one component inside an **SGML** or **XML** environment. It defines the syntactic rules according to which a document can be composed. There are no semantics associated with the elements and attributes defined in a **DTD**, although normally the names chosen for elements and attributes will have some meaning to them. Using a **DTD** and an **XML Processor**, a document can be validated against the **DTD**, which means it can be tested whether it conforms to a given **DTD**. **HTML** is one example of an **SGML DTD** (which has been reformulated as an **XML DTD** in **XHTML**).

## Dublin Core

The **Dublin Core** is a **Meta Data** element set intended to facilitate discovery of electronic resources. Originally conceived for author-generated description of **WWW** resources, it has attracted the attention of formal resource description communities such as museums, libraries, government agencies, and commercial organizations. One example application of the **Dublin Core** elements is OEB.

## EBNF - Extended Backus-Naur Form

An **EBNF** is any variation on the basic **BNF** meta-syntax notation with (some of) the following additional constructs: square brackets surrounding optional items, asterisk suffix for a sequence of zero or more of an item, plus-sign suffix for one or more of an item, curly brackets enclosing a list of alternatives, and super- and subscripts indicating the number of possible occurrences. All these constructs can be expressed in plain **BNF** using extra productions and have been added for readability and succinctness.

## ECMA - European Computer Manufacturers Association

**ECMA** is an international, europe-based industry association founded in 1961 and dedicated to the standardization of information and communication systems. Many **ECMA** standards have been accepted as a base for international and european standards. To ensure close cooperation **ECMA** has established formal liaisons with all european and international standardization bodies. **ECMA** standards are developed by highly qualified experts from information technology and telecommunication industry with the commitment to provide in a consensus mode technical solutions ready for implementation in product development and conformance testing.

## ECMAScript

**ECMAScript** is the standardized version of the scripting language originally known as **JavaScript**. After the standardization of the language by **ECMA**, references to the language itself should use the term **ECMAScript**, while **JavaScript** and **JScript** are two implementations of this language.

## ECML - Electronic Commerce Markup Language

**ECML** provides a set of simple guidelines for **WWW** merchants that will enable electronic wallets from multiple vendors to fill in their HTML Forms. **ECML** defines a number of simple field types for B2C scenarios, which should be used by HTML Forms for naming fields. Using these standardized fields, mechanisms on the **Browser** side (such as electronic wallets) can automatically identify and fill out forms for the user, making transactions for the user more easy.

## EDI - Electronic Data Interchange

**EDI** is the computer-to-computer exchange of business data in standard formats. In **EDI**, information is organized according to a specified format set by both parties, allowing an automated computer transaction that requires no human intervention or rekeying on either end. The information contained in an **EDI** transaction set is, for the most part, the same as on a conventionally printed document. **EDI** standards in the US are set and published by the ASC X12 committee of **ANSI**. International EDI standards are known as EDIFACT standards.

## EGA - Enhanced Graphics Adapter

A **Display Standard** referring to a video adapter capable of a resolution of up to 640 by 350 pixels.

### Email Protocol

The [Email Protocol](#) is a base topic for topics describing protocols for implementing Email services.

### ESMTP - Extended SMTP

[ESMTP](#) describes a framework for extensions of [SMTP](#). [SMTP](#) still provides the basic mechanism for exchanging Email messages, but it has become apparent that it lacks some important functionality. Instead of defining a revised but static new version of [SMTP](#), [ESMTP](#) defines a mechanism how extensions can be integrated into the basic [Protocol](#), and how these extensions can be used in an interoperable way.

### EXIF - Exchangeable Image File Format

Many digital cameras store images using [EXIF](#) compressed files. [EXIF](#) compressed files use [JPEG](#) for [Image Compression](#). This means the image data can be read by any application supporting [JPEG](#). In addition, [EXIF](#) stores [Meta Data](#) within application segments at the beginning of the file, and uses [sRGB](#) as the default color space. It is recommended that [EXIF](#) image files should be named and arranged in directories according to the DCF specification.

### FAQ - Frequently Asked Questions

[FAQs](#) are collections of questions and answers that frequently occur for a specific subject. [FAQ](#) have a long tradition in [Usenet News](#) newsgroups, where they are periodically posted for bringing new subscribers up-to-date.

### FastCGI

[FastCGI](#) is an extension of the [CGI](#) which eliminates its drawbacks and provides high performance, while remaining highly compatible with existing [CGI](#) applications. [FastCGI](#) is conceptually very similar to [CGI](#), with two major differences. As the first difference, [FastCGI](#) processes are persistent, after finishing a request, they wait for a new request instead of exiting. The second difference is that, instead of using [Operating System](#) environment variables and pipes, the [FastCGI](#) protocol multiplexes the environment information, standard input, output and error over a single full-duplex connection. This allows [FastCGI](#) programs to run on remote machines, using [TCP](#) connections between the [HTTP Server](#) and the [FastCGI](#) application.

### File System

The [File System](#) is a base topic for topics describing systems for the structured storage of individual files.

### Firewall

A [Firewall](#) is a special kind of [Gateway](#) that is used to block certain types of network traffic (typically, network traffic that is potentially dangerous). Most [Firewalls](#) work as packet fil-

ters, comparing the passing packets to a set of filter rules. Typically, the comparison performed by the packet filter involves the source address, the source [Port](#), the destination address, and the destination [Port](#). Filtering on source and destination addresses grants control over who may communicate with the internal network. All traffic from undesirable networks can be screened out. [Ports](#), on the other hand, are used to distinguish network services. By filtering out a [Port](#), it is possible to deny the outside world access to a service offered on the internal network.

### Font Format

The [Font Format](#) is a base topic for topics describing a defined way of coding information for describing fonts (ie, collections of glyphs for displaying characters).

### Format

The [Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer.

### FQDN - Fully Qualified Domain Name

A [FQDN](#) is a domain name that includes all higher level domains relevant to the entity named.

### FQHN - Fully Qualified Host Name

A [FQHN](#) is either the [FQDN](#) of a host (ie, a completely specified domain name ending in a [TLD](#)), or the numeric IP address of a host.

### Framework

The [Framework](#) is a base topic for topics describing a software product which can be used to run or construct applications.

### FTP - File Transfer Protocol

[FTP](#) is a [Protocol](#) for file transfer between hosts on the [Internet](#). The primary function of [FTP](#) is to transfer files efficiently and reliably among hosts and to allow the convenient use of remote file storage capabilities. The objectives of [FTP](#) are to promote sharing of files (computer programs and/or data), to encourage indirect or implicit (via programs) use of remote computers, to shield users from variations in file storage systems among hosts, and to transfer data reliably and efficiently. [FTP](#), though usable directly by a user at a terminal, is designed mainly for use by programs.

### FTP Client

The [FTP Client](#) is a base topic for topics describing clients implementing [FTP](#). A [FTP Client](#) communicates with a [FTP Server](#).

### FTP Server

The [FTP Server](#) is a base topic for topics describing servers implementing [FTP](#). A [FTP Server](#) communicates with a [FTP Client](#).

### Gateway

A [Gateway](#) is a device that connects networks segments on any layer higher than the network layer of the network (typically on the application layer). Related to [Gateways](#) are [Repeaters](#), [Bridges](#), and [Routers](#), which also connect network segments, but on different layers of the networking architecture.

### GIF - Graphics Interchange Format

[GIF](#) is a data stream-oriented [Image Format](#) used to define the transmission protocol of [LZW](#)-encoded bitmap data. [GIF](#) images may be up to eight bits (256 colors) in depth and are always compressed. Despite the fact that [GIF](#) supports only 8-bits worth of colors, and the multimedia extensions introduced in the [GIF89a](#) release have not been widely utilized, [GIF](#) still remains a popular choice for storing lower resolution image data. Any software created or modified after 1994 that supports the capability of reading and/or writing [GIF](#) files must obtain a patent license agreement from Unisys Corporation. For publishing on the [WWW](#), [JPEG](#) is a good companion format for photo-realistic images, while [PNG](#) has been designed to replace [GIF](#) in the long run.

### GIOP - General Inter-ORB Protocol

[GIOP](#) is the abstract [Protocol](#) which is used for communications between [CORBA ORBs](#). It specifies the transfer syntax and a standard set of message formats for [ORB](#) interoperation over any connection-oriented transport [Protocol](#). [GIOP](#) is designed to be simple and easy to implement, while still allowing for reasonable scalability and performance.

### Gopher

The [Gopher Protocol](#) is designed primarily to act as a distributed document delivery system. While documents (and services) reside on many [Servers](#), [Gopher Client](#) software presents users with a hierarchy of items and directories much like a file system. In fact, the [Gopher Interface](#) is designed to resemble a file system since a file system is a good model for locating documents and services. The user sees what amounts to one big networked information system containing primarily document items, directory items, and search items (the latter allowing searches for documents across subsets of the information base). Since the [WWW](#) allows greater flexibility in the structure and presentation of distributed information, the usage of gopher services and the number of gopher servers is getting smaller.

### GPRS - General Packet Radio Service

[GPRS](#) represents the first implementation of packet switching within The [GSM](#), which is essentially a circuit-switched technology. Using [GPRS](#) will enable users to send and receive data at speeds up to 115kbit/s. [GPRS](#) is very efficient in its use of scarce spectrum resources

and enables [GPRS](#) operators to introduce a wide range of value added services. [GPRS](#) is ideal for bursty data applications such as Email or [Internet](#) access, and can also enable "virtual permanent connections" to data sources, allowing information to arrive rather than being sought. This cannot be achieved using standard circuit-switched networks.

### Graphics Format

The [Graphics Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer of graphics.

### Group

The [Group](#) is a base topic for topics describing a group of bodies.

### GSM - Global System for Mobile Communications

[GSM](#) is a digital cellular communications system. It was developed in order to create a common european mobile telephone standard but it has been rapidly accepted worldwide. [GSM](#) was designed to be compatible with ISDN services. The disadvantage of this design is the focus on circuit-switched technologies, which currently makes [GSM](#) less than ideally suited for mobile data access. However, with the introduction of [GPRS](#) in newer versions of the [GSM](#) standard, [GSM](#) becomes capable of handling packet-switched traffic.

### gTLD - Generic TLD

A [gTLD](#) is a [DNS TLD](#) which is not specific to a country. Each of the [gTLDs](#) was created for a general category of organizations. Generally, under the [gTLDs](#) the structure is very flat. That is, many organizations are registered directly under the [gTLD](#), and any further structure is up to the individual organizations.

### GUI - Graphical User Interface

A [GUI](#) is the usual way to access a computer. The most common [GUIs](#) are the Windows [GUI](#), and the MacOS [GUI](#). [GUIs](#) usually employ a number of predefined design elements (such as pull-down menus and buttons) which are used to implement specific functions. The most common [Browsers](#) (Navigator and IE) also implement [GUIs](#), in this case for accessing the [WWW](#). Examples for non-[GUI Browsers](#) are text-based browsers such as Lynx, or entirely non-visual browsers (for example using speech synthesis techniques).

### Handle System

The [Handle System](#) is a comprehensive system for assigning, managing, and resolving persistent [Identifiers](#), known as "handles", for digital objects and other resources on the Internet. Handles can be used as [URNs](#). The [Handle System](#) includes an open set of [Protocols](#), a name space, and an implementation of the [Protocols](#). The [Protocols](#) enable a distributed computer system to store handles of digital resources and resolve those handles into the information necessary to locate and access the resources. This associated information can be changed as needed to reflect the current state of the identified resource without changing the handle, thus

allowing the name of the item to persist over changes of location and other state information. Combined with a centrally administered naming authority registration service, the [Handle System](#) provides a general purpose, distributed global naming service for the reliable management of information on networks over long periods of time.

### Hardware Interface

The [Hardware Interface](#) is a base topic for topics describing an [Interface](#) on the hardware level. This generally includes connectors (except for [Wireless Interfaces](#)) and physical transmission details, such as voltage and timing of signals.

### High Sierra

The [High Sierra](#) format is a logical format for CD-ROM media. It was used mainly between 1986 and 1988, after 1988 it was replaced by the [ISO 9660](#) standard for the logical format on CD-ROM media ([High Sierra](#) and [ISO 9660](#) are identical in content, but the exact format is different).

### HTML - Hypertext Markup Language

[HTML](#) is a simple markup language used to create hypertext documents that are platform independent. [HTML](#) documents are [SGML](#) documents with generic semantics that are appropriate for representing information from a wide range of domains. For example, [HTML](#) markup can represent [Usenet News](#), Email, documentation, menus of options, database query results, simple structured documents with in-lined images, and hypertext views of existing bodies of information.

### HTTP - Hypertext Transfer Protocol

[HTTP](#) is the protocol used for information exchange on the [WWW](#). [HTTP](#) defines how messages are formatted and transmitted, and what actions a [HTTP Server](#) and an [HTTP Client](#) (which in most cases is a [Browser](#)) should take in response to various messages. [HTTP](#) uses a reliable, connection-oriented transport service such as the [TCP](#). [HTTP](#) is a stateless [Protocol](#), where each request is interpreted independently, without any knowledge of the requests that came before it.

### HTTP Client

The [HTTP Client](#) is a base topic for topics describing clients implementing [HTTP](#). An [HTTP Client](#) communicates with an [HTTP Server](#).

### HTTP Extension Framework

[HTTP](#) is being used for an increasing number of applications involving distributed authoring, collaboration, printing, and various RPC-like [Protocols](#). The [HTTP Extension Framework](#) is an extension mechanism for [HTTP](#) designed to address the tension between private agreement and public specification, and to accommodate extension of [HTTP Clients](#) and [HTTP Servers](#) by software components.

### HTTP Proxy

In the context of the [WWW](#), an [HTTP Proxy](#) is an intermediary program which acts as both an [HTTP Server](#) and an [HTTP Client](#), receiving a request from a [Client](#) (in most cases a [Browser](#)) and then acting as an [HTTP Client](#) and making requests on behalf of other [HTTP Clients](#). However, requests to an [HTTP Proxy](#) can also be serviced internally, for example if the [HTTP Proxy](#) uses its Cache instead of sending a request to the origin [HTTP Server](#). In order to use an [HTTP Proxy](#), the [HTTP Client](#)'s request has to be explicitly addressed to the [HTTP Proxy](#), which then sends a request to the origin [HTTP Server](#). An [HTTP Proxy](#) may also perform a protocol conversion, for example a [Browser](#) may send an [HTTP](#) request to the [HTTP Proxy](#) referencing an [FTP](#) resource, and the [HTTP Proxy](#) then acts as [FTP Client](#) to retrieve the resource from the [FTP Server](#) using [FTP](#), and eventually the resource is sent back from the [HTTP Proxy](#) to the [Browser](#) using [HTTP](#).

### HTTP Server - Web Server

The [HTTP Server](#) is a base topic for topics describing servers implementing [HTTP](#) (aka [WWW](#) servers or Web servers). An [HTTP Server](#) communicates with an [HTTP Client](#).

### HTTPS - HTTP over SSL

[HTTPS](#) is a secure way of using [HTTP](#). [HTTP](#) provides almost no security features, it contains only basic authentication mechanisms, and no support for privacy. [HTTPS](#) solves this problem by replacing [HTTP](#)'s transport layer, the insecure [TCP](#), with [SSL](#), a secure transport layer. In the near future, [SSL](#) will probably be replaced by the more general [TLS](#) protocol, but it is very unlikely that the already established name of [HTTPS](#) will be changed to reflect this change.

### Hypermedia Format

The [Hypermedia Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer of hypermedia (usually used in a [Hypermedia System](#)).

### Hypermedia System

A [Hypermedia System](#) is an application which uses associative relationships amongst information contained within multiple media data for the purpose of facilitating access to, and manipulation of, the information encapsulated by the data.

### Hypertext Format

The [Hypertext Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer of hypertext (which is a special form of a [Hypermedia Format](#) restricted to textual information).

### I18N - Internationalization

[I18N](#) is the process of designing an application so that it can be adapted to various languages

and regions without engineering changes. The abbreviation is based on the fact that the rather lengthy word 'internationalization' starts with an 'i' followed by 18 characters followed by an 'n'. An internationalized system has the following characteristics: after [L10N](#), the same executable can run worldwide; textual elements, such as status messages and the GUI component labels, are not hardcoded in the program (instead they are stored outside the source code and retrieved dynamically); support for new languages does not require recompilation; culturally-dependent data, such as dates and currencies, appear in formats that conform to the end user's region and language; [L10N](#) can be done quickly.

## IAB - Internet Architecture Board

The [IAB](#) is the technical body that oversees the development of the [Internet Protocol](#) suite. It has two task forces, the [IETF](#) and the [IRTF](#).

## ICMP - Internet Control Message Protocol

[ICMP](#) is at the same [Protocol](#) layer as [IP](#), its purpose is to transmit information needed to control [IP](#) traffic. It is used mainly to provide information about routes to destination addresses. [ICMP](#) redirect messages inform hosts about more accurate routes to other systems, whereas [ICMP](#) unreachable messages indicate problems with a route. Additionally, [ICMP](#) can cause [TCP](#) connections to terminate gracefully if the route becomes unavailable. The ping command is a commonly-used [ICMP](#)-based service.

## IDEA - International Data Encryption Algorithm

[IDEA](#) is a symmetric block cipher. The speed of [IDEA](#) in software is similar to that of [DES](#). One of the principles during the design of [IDEA](#) was to facilitate analysis of its strength against differential cryptanalysis. [IDEA](#) is considered to be immune from differential cryptanalysis. In addition, no linear cryptanalytic attacks on [IDEA](#) have been reported and there is no known algebraic weakness in [IDEA](#).

## Identifier

The [Identifier](#) is a base topic for topics describing ways for identifying resources.

## IETF - Internet Engineering Task Force

[IETF](#) is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the [Internet](#) architecture and the smooth operation of the [Internet](#). It is open to any interested individual. The actual technical work of the [IETF](#) is done in its working groups, which are organized by topic into several areas (eg, routing, transport, security, etc.).

## IIOP - Internet Inter-ORB Protocol

[IIOP](#) is the most commonly used [Protocol](#) for communications in [CORBA](#). [IIOP](#) is a [Protocol](#) for the [Client/Server-Model](#) between two [CORBA](#) implementations. In a similar way to [HTTP](#), which uses [URIs](#) to locate [Servers](#) and in requests from [Clients](#) to [Servers](#), [CORBA](#) uses

an [IOR](#) for identifying remote objects. [IORs](#) can be used to invoke operations on remote [CORBA](#) systems, using [IIOP](#) for communications.

### Image Compression

The [Image Compression](#) is a base topic for topics describing a defined way of compressing images.

### Image Format

The [Image Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer of images.

### IMAP - Internet Message Access Protocol

[IMAP](#) is a [Email Protocol](#) allowing a [Client](#) to access and manipulate Email messages on a [Server](#). It permits manipulation of remote message folders (mailboxes) in a way that is functionally equivalent to local mailboxes. [IMAP](#) includes operations for creating, deleting, and renaming mailboxes; checking for new messages; permanently removing messages; searching; and selective fetching of message attributes, texts, and portions thereof. It does not specify a means of posting mail; this function is handled by a mail transfer protocol such as [SMTP](#). [IMAP](#) is newer and more powerful than [POP](#), which is used for the same application area.

### Instant Messaging System

The [Instant Messaging System](#) is a base topic for topics describing a system for the synchronous exchange of messages.

### Interface

The [Interface](#) is a base topic for topics describing an [Interface](#), which is a clear separation between different entities.

### International Organization

The [International Organization](#) is a base topic for topics describing a multi-Governmentally influenced body.

### Internet

The [Internet](#) is the entirety of all computers which are interconnected (using various physical [Networking](#) techniques) and employ the [Internet Protocol](#) suite on top of their networking systems.

### IOR - Interoperable Object Reference

An [IOR](#) is a data structure that stores information needed to locate and communicate with a [CORBA](#) object over one or more [Protocols](#). For example, an [IOR](#) containing [IIOP](#) informa-

tion stores IP address and [TCP Port](#) number information.

## IP - Internet Protocol

[IP](#) is specifically limited in scope to provide the functions necessary to deliver a package of bits (an [Internet](#) datagram) from a source to a destination over an interconnected system of networks. There are no mechanisms to augment end-to-end data reliability, flow control, sequencing, or other services commonly found in host-to-host [Protocols](#). In most cases, [TCP](#) is used on top of [IP](#).

## IPng - Internet Protocol next generation

Although [IP](#) is the most visible [Protocol](#) of the [Internet](#), there are many other [Protocols](#) which are also part of the [Internet](#) architecture and which also have to be changed when making the transition from IPv4 to [IPv6](#). In the context of [IPv6](#), there are also a number of protocols which are new to the [Internet](#) architecture. The common practice is to use the term [IPng](#) to refer to all protocols which have to be changed or added when switching to [IPv6](#).

## IPR - Intellectual Property Rights

[IPR](#) are temporary grants of monopoly intended to give economic incentives for innovative activity. [IPR](#) exist in the form of patents, copyrights, and trademarks.

## IPv6 - Internet Protocol Version 6

[IPv6](#) is a new version of [IP](#) which is designed to be an evolutionary step from IPv4 (commonly referred to simply as [IP](#)). It can be installed as a normal software upgrade in [Internet](#) devices and is interoperable with the current IPv4. Its deployment strategy was designed to not have any "flag" days. [IPv6](#) is designed to run well on high performance networks (eg, ATM) and at the same time is still efficient for low bandwidth networks (eg, [Wireless Interfaces](#)). In addition, it provides a platform for new [Internet](#) functionality (such as resource reservation capabilities) that will be required in the near future.

## IRI - Internationalized Resource Identifier

An [IRI](#) is a generalized form of [URI](#) that may contain non-[ASCII](#) characters.

## IRTF - Internet Research Task Force

The [IRTF](#) is composed of a number of focused, long-term and small research groups. These groups work on topics related to [Internet Protocols](#), applications, architecture and technology. The [IRTF](#) focuses on longer term research issues related to the [Internet](#), while the parallel organization, the [IETF](#), focuses on the shorter term issues of engineering and standards making.

## ISBN - International Standard Book Number

The [ISBN](#) is a nine- or ten-digit number which identifies a specific book title and is recognized worldwide. Publishers assign [ISBNs](#) to books to simplify and expedite their ordering

and purchase. An [ISBN](#) number can often be found on the back of the title page of a book or at the bottom of the back cover, though many older books do not have [ISBNs](#). Sometimes more than one book has the same [ISBN](#), as is frequently the case with books in a series. Book records for books in a series may also contain an [ISSN](#) for the series. Separate [ISBNs](#) are usually assigned for each format of a book (such as hard-cover or paperback). Newer books frequently also have an EAN-encoded version of the [ISBN](#) on their cover.

## ISO 9660

The [ISO 9660](#) format is a logical format for CD-ROM media. It was standardized in 1988 and replaced the [High Sierra](#) standard for the logical format on CD-ROM media ([ISO 9660](#) and [High Sierra](#) are identical in content, but the exact format is different).

## ISSN - International Standard Serial Number

The [ISSN](#) is an internationally accepted code which identifies serial publications. It is an eight-digit number consisting of seven digits plus a check digit which enables a computer to recognize when the number is incorrectly cited. The check digit may be an "X", otherwise the [ISSN](#) is fully numeric. An [ISSN](#) may be used as control numbers for serial titles in automated systems, for example for identifying titles, ordering, checking in, and claiming by libraries and subscription agents. It may be used to ensure more accurate serials citation by scholars, researchers, abstractors and librarians and is particularly helpful when distinguishing between serials with identical titles. An increasingly important use of the [ISSN](#) is as a component in EAN barcodes for magazines.

## ITU - International Telecommunication Union

[ITU](#) is an [International Organization](#), within which the public and private sectors cooperate for the development of telecommunications. [ITU](#) adopts international regulations and treaties governing all terrestrial and space uses of the frequency spectrum as well as the use of the geostationary-satellite orbit, within which countries adopt their national legislation. It also develops standards to facilitate the interconnection of telecommunication systems on a world-wide scale regardless of the type of technology used.

## JAR - Java Archive

[JAR](#) is a platform-independent [Format](#) that aggregates many files into one. Multiple [Java applets](#) and their requisite components (class files, images, and sounds) can be bundled in a [JAR](#) file and subsequently downloaded to a [Browser](#) in a single [HTTP](#) transaction, improving the download speed. The [JAR](#) format also supports compression, which reduces the file size, further improving the download time. In addition, the [applet](#) author can digitally sign individual entries in a [JAR](#) file to authenticate their origin.

## Java

[Java](#) is a general-purpose object-oriented [Programming Language](#). [Java](#) is interesting in the context of the [WWW](#) because it is compiled into [Java](#) "bytecode", which is executed on a [JVM](#). This design makes [Java](#) programs platform-independent, and [Java applets](#), a special

form of [Java](#) programs, can be integrated into [WWW](#) documents. Most [Browsers](#) today contain a [JVM](#) and a run-time environment for [applets](#).

## JavaScript

[JavaScript](#) is a scripting language designed to be used for [WWW](#) pages. Scripts are embedded within [WWW](#) pages, and [Browsers](#) interpret these scripts after loading the page. [JavaScript](#) has a syntax similar to [Java](#), but it is an entirely different and less powerful language. [JavaScript](#) has been the source for the standardization of [ECMAScript](#).

## JAXP - Java API for XML Parsing

[JAXP](#) is an [API](#) for [XML](#) parsing and processing. It is an abstraction layer (ie, an [API](#) for other [APIs](#)) which can be used to access [XML](#)-specific functionality from within [Java](#) programs. [JAXP](#) supports parsing [XML](#) documents using the [DOM](#) or [SAX](#) [APIs](#), and processing [XML](#) documents with [XSLT](#) using the [TrAX](#) [API](#).

## JBIG - Joint Bi-level Image Experts Group

[JBIG](#) is the name of the committee that designed the [Image Compression](#) algorithm that is also called [JBIG](#).

## JBIG - Joint Bi-level Image Experts Group

[JBIG](#) is an [Image Compression](#) standard that is mainly intended as an improvement of [ITU's](#) G3 Fax and G4 Fax recommendations for facsimile transmission. Apart from coding bilevel (ie, black and white) images, [JBIG](#) can also be used for coding grayscale and color images with limited numbers of bits per pixel. It uses a lossless compression algorithm which typically reduces the size of the uncompressed image by a factor of twenty to one. [JBIG](#) only defines an [Image Compression](#), not an [Image Format](#). [JBIG](#) compressed data is often stored in [TIFF](#) files.

## JDOM

[JDOM](#) is an [API](#) for [XML](#) documents. Unlike [DOM](#), which has been designed to be used in a variety of [Programming Languages](#), [JDOM](#) has been created specifically for the [Java Programming Language](#). Thus, [JDOM](#) has been optimized for [Java](#), and is easier to use than [DOM](#). [JDOM](#) is not based on [DOM](#), and it can be used independently from the underlying [XML Processor](#).

## JFIF - JPEG File Interchange Format

[JFIF](#) is the technical name for the [Image Format](#) better (but inaccurately) known as [JPEG](#). This term is used only when the difference between the [Image Format](#) and the [Image Compression](#) is crucial. Strictly speaking, however, [JPEG](#) does not define an [Image Format](#), and therefore in most cases it would be more precise to speak of [JFIF](#) rather than [JPEG](#). Another [Image Format](#) for [JPEG](#) is [SPIFF](#) defined by the [JPEG](#) standard itself, but [JFIF](#) is much more widespread than [SPIFF](#).

## JPEG - Joint Photographic Experts Group

**JPEG** is the name of the committee that designed the [Image Compression](#) algorithm that is also called **JPEG**.

## JPEG - Joint Photographic Experts Group

**JPEG** is an [Image Compression](#) algorithm that is designed for compressing either full-color or grayscale digital images of natural, real-world scenes. It does not work very well on non-realistic images, such as cartoons or line drawings. **JPEG** does not handle compression of black and white (1 bit per pixel) images or moving pictures. **JPEG** itself does not describe an [Image Format](#), it only specifies the compression algorithm. Some [Image Formats](#) for exchanging images compressed with the **JPEG** algorithm are the wide-spread **JFIF** and the less popular **SPIFF**.

## JScript

Microsoft's **JScript** scripting language is a superset of the standardized [ECMAScript](#) scripting language. It is mainly intended to be used as a scripting language on [HTML](#) pages. Scripts are embedded within [WWW](#) pages, and [Browsers](#) interpret these scripts after loading the page.

## JSP - Java Server Pages

Sun Microsystems's **JSP** technology uses [XML](#)-like tags and scriptlets written in [Java](#) to encapsulate the logic that generates the content for a [HTML](#) or [XML](#) page. Additionally, the application logic can reside in server-based resources that the page accesses with these tags and scriptlets. Any and all formatting tags ([HTML](#) or [XML](#)) are passed directly back to the response page. This separates the page logic from its design and display, and thereby supports a reusable component-based design. **JSP** is an extension of the [Java](#) servlet [API](#). **JSP** is Sun Microsystems's proprietary technology to compete with Microsoft's [ASP](#).

## JVM - Java Virtual Machine

The **JVM** is the "processor architecture" on which [Java](#) bytecode (a compiled [Java](#) program) is executed. In contrast to other processor architectures, **JVM** is mostly implemented in software, and available for a large variety of hardware platforms, ranging from mainframe computers to mobile phones and small embedded devices.

## L10N - Localization

**L10N** is the process of adapting software for a specific region or language by adding locale-specific components and translating text. The abbreviation is based on the fact that the rather lengthy word 'localization' starts with a 'l' followed by 10 characters followed by an 'n'. Usually, the most time-consuming portion of **L10N** is the translation of text. Other types of data, such as sounds and images, may require **L10N** if they are culturally sensitive. The formatting of dates, numbers, and currencies also must conform to local requirements. The effort required for the **L10N** of a system heavily depends on how much **I18N** has been taken into account when building the system.

## LAMP - Linux/Apache/MySQL/PHP

**LAMP** is a concept (or a philosophy) describing an open-source **WWW** development platform. **LAMP** is fully based on open-source components, using Linux as its **Operating System**, the Apache HTTP Server as its **HTTP Server**, MySQL as its RDBMS, and **PHP** as its **Server Side Technology**. Sometimes, the 'P' in **LAMP** is interpreted as meaning **Python** or **Perl**, but in most cases it refers to **PHP**.

## LaTeX

**LaTeX** is a macro package based on the **TeX** typesetting system. **LaTeX** provides macros for the most frequently used document concepts, such as sectioning, lists, or tables.

## LDAP - Lightweight Directory Access Protocol

**LDAP** was defined in order to encourage adoption of **X.500** directories. **DAP** was regarded as being too complex for simple **Internet Clients** to use. **LDAP** defines a relatively simple **Protocol** for updating and searching directories running over **TCP**.

## LDH - Letter Digit Hyphen

**LDH** form is a **Format** for host names in the **DNS**. It constrains valid names to contain only **ASCII** letters, digits, and hyphens.

## LDIF - LDAP Data Interchange Format

**LDIF** is typically used to import and export directory information between **LDAP**-based directory servers, or to describe a set of changes which are to be applied to a directory. **LDIF** is also frequently used by Email agents as file format for directory information (such as address books). **LDIF** is a text-based format which can be easily manipulated using text-based tools (for **XML**-based applications, **DSML** is available as an **XML**-based version for **LDAP** information).

## Local File System

The **Local File System** is a base topic for topics describing local file systems.

## LZ77 - Lempel-Ziv

**LZ77** is a data **Compression** algorithm that builds a dictionary of frequently repeated groups of bit patterns on a per-file basis and represents these frequent patterns as shorter bit patterns (using the dictionary).

## LZW - Lempel-Ziv-Welch

**LZW** is a refinement of the **LZ77** algorithm. The **LZW** algorithm is patented by Unisys Corporation.

## MAC - Message Authentication Code

A [MAC](#) is an authentication tag (also called a checksum) derived by application of an authentication scheme, together with a secret key, to a message. [MACs](#) are computed and verified with the same key so they can only be verified by the intended receiver, unlike digital signatures. [MACs](#) can be derived from various cryptographic techniques and algorithms, such as [SHA](#), [MD5](#), or [DES](#).

## Magnetic Storage Medium

The [Magnetic Storage Medium](#) is a base topic for topics describing media for magnetic data storage.

## Magneto-Optical Storage Medium

The [Magneto-Optical Storage Medium](#) is a base topic for topics describing media for magneto-optical data storage.

## MCF - Meta Content Framework

[MCF](#) is a proposal authored by Netscape Communications for the definition of [WWW Meta Data](#). The proposal has been used as input for [W3C](#)'s work on [RDF](#).

## MD2 - Message Digest 2

The [MD2](#) algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given pre-specified target message digest. [MD2](#) is optimized for 8-bit machines and intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private key under a [Public-Key Cryptography](#) system such as [RSA](#).

## MD4 - Message Digest 4

The [MD4](#) algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given pre-specified target message digest. [MD4](#) is aimed at 32-bit machines and intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private key under a [Public-Key Cryptography](#) system such as [RSA](#).

## MD5 - Message Digest 5

The [MD5](#) algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any

message having a given pre-specified target message digest. [MD5](#) is aimed at 32-bit machines and intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private key under a [Public-Key Cryptography](#) system such as [RSA](#).

### Meta Data

[Meta Data](#) is data about data. Since the term "data" is general in the sense that it may also be [Meta Data](#), there is no such thing as "meta meta data". [RDF](#) is an attempt to define a general and machine understandable format for [WWW Meta Data](#). The benefits of machine understandable [Meta Data](#) are more powerful ways of information search, retrieval, and processing.

### MIDI - Musical Instrument Digital Interface

[MIDI](#) enables people to use multimedia computers and electronic musical instruments. There are actually three components to [MIDI](#), the communications "[Protocol](#)", the [Hardware Interface](#) and a distribution format called "Standard MIDI Files". In the context of the [WWW](#), the most interesting component is the [Audio Format](#). In principle, [MIDI](#) files contain sequences of [MIDI Protocol](#) messages. However, when [MIDI Protocol](#) messages are stored in [MIDI](#) files, the events are also time-stamped for playback in the proper sequence. Music delivered by [MIDI](#) files is the most common use of [MIDI](#) today.

### MIME - Multipurpose Internet Mail Extensions

[MIME](#) provide facilities to allow multiple objects in a single [Internet](#) Email message, to represent body text in character sets other than [ASCII](#), to represent formatted multi-font text messages, to represent non-textual material such as images and audio fragments, and generally to facilitate later extensions defining new types of [Internet](#) Email for use by cooperating email agents.

### Modem - Modulation/Demodulation

A [Modem](#) converts digital signals into analog signals which can be transmitted over an analog line (in many cases a [PSTN](#) connection), and transforms incoming analog signals into their digital equivalents. The specific technique used to encode the digital bits into analog signals is called "[Modulation Protocol](#)", which defines the exact method of encoding and the data transfer speed. A [Modem](#) typically supports more than one [Modulation Protocol](#).

### Modulation Protocol

The [Modulation Protocol](#) is a base topic for topics describing protocols for modulating and demodulating digital signals over analog connections.

### MOSS - MIME Object Security Services

[MOSS](#) is a [Protocol](#) that uses the [MIME](#) multipart/signed and multipart/encrypted framework to apply digital signature and encryption services to [MIME](#) objects. The services are offered through the use of end-to-end cryptography between an originator and a recipient at the ap-

plication layer. [Public-Key Cryptography](#) is used in support of the digital signature service and encryption key management. [Secret-Key Cryptography](#) is used in support of the encryption service. The procedures are intended to be compatible with a wide range of public key management approaches, including both ad hoc and certificate-based schemes. Mechanisms are provided to support many public key management approaches.

### Movie Sound Format

The [Movie Sound Format](#) is a base topic for topics describing a [Format](#) for movie soundtracks.

### Mozilla

In March 1998, Netscape Communications decided that the Communicator product (including the Navigator [Browser](#)) would be available free of charge, and that the source code would also be available free of charge. The idea behind this decision is to encourage the public to take part in the development of Navigator (which has the nickname of [Mozilla](#)). Hopefully, this will result in more functionality and availability for more platforms than could be provided by Netscape Communications alone. The [Mozilla](#) organization is hosted by Netscape Communications and tries to coordinate the efforts by managing the source code and maintaining a list of what should be implemented in the future.

### National Organization

The [National Organization](#) is a base topic for topics describing a single governmentally influenced [Organization](#).

### Network

The [Network](#) is a base topic for topics describing networks for transmitting signals.

### NITF - News Industry Text Format

[NITF](#) uses [XML](#) to define the structure and content of news articles (text and statistical data). It identifies structural pieces such as headlines, bylines, paragraphs, table columns and footnotes. The subjects covered by the content may be indicated through the use of IPTC subject codes. Rich in-line markup can be applied to specify such things as organizations, events, places and people. Because metadata tags are applied throughout the news content, [NITF](#) documents are more searchable and useful than [HTML](#) pages. [NITF](#) documents, like other news data may be contained within, or referenced from, a NewsML wrapper.

### NNTP - Network News Transport Protocol

[NNTP](#) specifies a protocol for the distribution, inquiry, retrieval, and posting of [Usenet News](#) articles using a reliable stream-based transmission of [Usenet News](#) among the [Internet](#) community. [NNTP](#) is designed so that [Usenet News](#) articles are stored in a central database allowing a subscriber to select only those items he wishes to read. Indexing, cross-referencing, and expiration of aged messages are also provided.

## Nonce

A [Nonce](#) is a randomly generated value used to defeat "playback" attacks in communication [Protocols](#). One party randomly generates a nonce and sends it to the other party. The receiver encrypts it using the agreed upon secret key and returns it to the sender. Since the nonce was randomly generated by the sender, this defeats playback attacks because the replayer can not know in advance the nonce the sender will generate. The receiver denies connections that do not have the correctly encrypted nonce.

## NTP - Network Time Protocol

[NTP](#) provides the mechanisms to synchronize time and coordinate time distribution in a large, diverse [Internet](#) operating at rates from mundane to light-wave. It uses a returnable-time design in which a distributed subnet of time [Servers](#) operating in a self-organizing, hierarchical-master-slave configuration synchronizes local clocks within the subnet and to national time standards via wire or radio. The [Servers](#) can also redistribute reference time via local routing algorithms and time [Servers](#). A simpler variant of [NTP](#) has been specified under the name of [SNTP](#).

## NUN - Normalized Universal Name

A [NUN](#) is a name that uniquely identifies an element, attribute, simple type, complex type, attribute group, model group, or notation declaration in an [XML Schema](#).

## OASIS - Organization for the Advancement of Structured Information Standards

[OASIS](#) is a nonprofit, international [Consortium](#) dedicated to accelerating the adoption of product-independent formats based on public standards. These standards include [SGML](#), [XML](#), and [HTML](#) as well as others that are related to structured information processing. Members of [OASIS](#) are providers, users, and specialists of the technologies that make these standards work in practice.

## OLE - Object Linking and Embedding

[OLE](#) is a compound document standard developed by Microsoft. [OLE](#) makes it possible to create objects with one application and link or embed them in a second application. Embedded objects retain their original format and links to the application that created them. Support for [OLE](#) is built into the Windows and MacOS [Operating Systems](#). A competing compound document standard developed mainly by Apple and IBM is called [OpenDoc](#).

## OMG - Object Management Group

Established in 1989, [OMG](#) promotes the theory and practice of object technology for the development of distributed computing systems. The goal is to provide a common architectural framework for object oriented applications based on widely available interface specifications. [OMG](#) has a membership of over 800 software vendors, software developers, and end users. [CORBA](#) is standardized by the [OMG](#).

### OpenDoc

[OpenDoc](#) is an open, multi-platform architecture for component software developed mainly by Apple and IBM. It is a [Component Model](#) as well as an [API](#) that makes it possible to design independent programs (components) that can work together on a single document. In favor of [Java](#) technology, Apple announced its plans to reduce its investment in [OpenDoc](#) technologies. The competing product to [OpenDoc](#) is Microsoft's [OLE](#).

### OpenType

The [OpenType Font Format](#) is an extension of the [TrueType Font Format](#), adding support for [Type 1](#) font data. The [OpenType Font Format](#) was developed jointly by Microsoft and Adobe. As with [TrueType](#) fonts, [OpenType](#) fonts allow the handling of large glyph sets using [Unicode](#) encoding. Such encoding allows broad international support, as well as support for typographic glyph variants. Additionally, [OpenType](#) fonts may contain digital signatures, allowing [Operating Systems](#) and [Browsers](#) to identify the source and integrity of font files, including embedded font files obtained in [WWW](#) documents, before using them. Also, font developers can encode embedding restrictions in [OpenType](#) fonts, and these restrictions cannot be altered in a font signed by the developer.

### Operating System

The [Operating System](#) is a base topic for topics describing operating systems.

### Optical Storage Medium

The [Optical Storage Medium](#) is a base topic for topics describing media for optical data storage.

### ORB - Object Request Broker

The [ORB](#) is the key component of the [CORBA](#) programming model. An [ORB](#) is responsible for transferring operations from [Clients](#) to [Servers](#). This requires the [ORB](#) to locate a [Server](#) implementation (and possibly activate it), transmit the operation and its parameters, and finally return the results back to the [Client](#).

### Organization

The [Organization](#) is a base topic for topics describing a organized group of bodies.

### OSI - Open Systems Interconnection

[OSI](#) is a reference model for computer communications. It was created as a competitor of the [Internet Protocols](#). While [OSI](#) as a whole has not been very successful, parts of it are widely used, and many concepts pioneered in [OSI](#) are slowly finding their way into [Internet](#) technologies. [OSI](#) is defined in terms of different "layers". [OSI](#)-based applications belong to the "application layer" (layer 7) of the [OSI](#) "Basic Reference Model". The application layer resides above the "presentation layer" (layer 6), which identifies alternative encodings, and the "ses-

sion layer" (layer 5), providing dialogue control. Collectively, these three layers provide application services, and are commonly referred to as the "upper layers". The "lower layers" of the [OSI](#) stack are "transport" (layer 4), "network" (layer 3), "link" (layer 2) and "physical" (layer 1).

## P2P - Peer to Peer

[P2P](#) networking is a network model where, depending on an operation's context, any node can operate as either a [Server](#) or a [Client](#). [P2P](#) provides certain interesting capabilities not possible in traditional [Client/Server-Model](#) networks, which have predefined [Server](#) or [Client](#) roles for their nodes.

## P3P - Platform for Privacy Preferences Project

The goal of [P3P](#) is to deal with the constant struggle between the need for [WWW](#) content providers to gain information about their readership and the need for these individuals to control the release of this information to others. [P3P](#) addresses the twin goals of meeting the data privacy expectations of consumers on the [WWW](#) while assuring that the medium remains available and productive for E-Commerce. Following the principle of providing consumers notice of site privacy policies, and allowing users to express and act upon their privacy preferences in a flexible manner, one goal enhances the success of the other.

## Parser

A [Parser](#) is a program for structuring a linear representation in accordance with a given grammar. For each grammar, there are generally an infinite number of linear representations (sentences) that can be structured with it. That is, a finite-size grammar can supply structure to an infinite number of sentences. This is the main strength of the grammar paradigm and indeed the main source of the importance of grammars: they summarize succinctly the structure of an infinite number of objects of a certain class. There are several reasons to perform this structuring process called parsing. The most important reason derives from the fact that the obtained structure helps to process the object further.

## Parser Generator

The [Parser Generator](#) is a base topic for topics describing tools for building [Parsers](#). Some tools are using separate components for lexical analysis and parsing (such as [lex/yacc](#)), others are integrated (such as [JavaCC](#)).

## PCT - Private Communication Technology

[PCT](#) was designed by Microsoft in an attempt to improve Netscape Communications's [SSL](#) protocol. However, development of [PCT](#) stopped when [SSL](#) was generalized as [IETF's TLS](#) protocol.

## PDF - Portable Document Format

[PDF](#) is a [Format](#) for representing documents in a manner that is independent of the original

application software, hardware, and operating system used to create those documents. A [PDF](#) file can describe documents containing any combination of text, graphics, and images in a device-independent and resolution independent format. The advantage of [PDF](#) over [PostScript](#) is the better accessibility of text within [PDF](#) files, which can still be searched for text and can also contain structural information like a table of contents.

## PDFLaTeX

[PDFLaTeX](#) is a special version of [LaTeX](#) which directly produces [PDF](#) output. Special commands in the [LaTeX](#) source can be used to control special [PDF](#) features, such as cross references or bookmarks.

## PDF/X

[PDF/X](#) is not an alternative to [PDF](#), it is a focused subset of [PDF](#) designed specifically for reliable prepress data interchange.

## PDOM - Persistent DOM

[PDOM](#) is a [Concept](#) rather than a specific [API](#) or [Format](#). While [DOM](#) defines a standardized [API](#) for accessing [XML](#) documents, [PDOM](#) is a concept for storing the in-memory information of the [DOM](#) tree on a persistent [Storage Medium](#). Typically, [PDOM](#) implementations use a binary and indexed [Format](#), and provide a [DOM-like API](#), which can be used to access this [Format](#). However, the specific [Format](#) used for [PDOM](#) varies and has not been standardized.

## Perl - Practical Extraction and Report Language

[Perl](#) is a general purpose interpreted [Programming Language](#), often used for scanning text and printing formatted reports. It provides extensive support for Regular Expression matching, dynamically scoped variables and functions, extensible run-time libraries, exception handling, and packages. [Perl](#) is frequently used for programming [CGI](#) applications.

## PGML - Precision Graphics Markup Language

[PGML](#) is a 2D [Graphics Format](#) meant to satisfy both the [WWW](#)'s scalable lightweight vector graphics needs and the precision needs of graphic artists who want to ensure that their graphic designs appear on end user systems with precisely the correct fonts, color, layout and compositing that they desire. [Graphics Format](#) uses the imaging model of [PostScript](#). [PGML](#) was a proposal authored by Adobe and has been used as input for the development of [SVG](#).

## PHP - PHP Hypertext Preprocessor

[PHP](#) is an [HTML](#)-embedded scripting language. Much of its syntax is borrowed from C, [Java](#), and [Perl](#), with the addition of a couple of unique [PHP](#)-specific features. The goal of the language is to allow [WWW](#) developers to write dynamically generated [HTML](#) pages quickly. [PHP](#) is either executed as a [CGI](#) script, or it is integrated into [HTTP Server](#) software, for example as an Apache HTTP Server module.

## PICS - Platform for Internet Content Selection

**PICS** is a pair of **Protocols**, allowing labels to be applied to **WWW** content. These **Protocols** empower any individual or organization to design and distribute labels reflecting their views about the content. **PICS** was pioneered by **W3C** as a practical alternative to global governmental censorship of the **Internet**. In addition, the same technology facilitates searching the **WWW** and provides a foundation for establishing trust in information on the **WWW**. **PICS** labels are rather limited in their expressiveness. A new version of **PICS** will be based on **RDF**, facilitating more complex labeling of resources.

## PIP - Presence Information Protocol

**PIP** enables the exchange of document based virtual presence information. Virtual presence information is the foundation for virtual neighborhood services which provide users with information about virtual neighbors, ie other users who are close within the virtual document space established by the **WWW**. **PIP** also allows the exchange of simple text-based messages.

## PKCS - Public-Key Cryptography Standards

**PKCS** are specifications produced by RSA Laboratories in cooperation with secure systems developers worldwide for the purpose of accelerating the deployment of **Public-Key Cryptography**. First published in 1991 as a result of meetings with a small group of early adopters of **Public-Key Cryptography** technology, the **PKCS** documents have become widely referenced and implemented. The **PKCS** standards cover **RSA** encryption, **Diffie-Hellman** key agreement, password-based encryption, extended-certificate syntax, **CMS**, private key information syntax, and certification request syntax, as well as selected attributes. Contributions from the **PKCS** series have become part of many formal and de facto standards, including **PKIX**, **SET**, **S/MIME**, and **SSL**.

## POP - Post Office Protocol

**POP** allows a **Client** computer to retrieve Email from a **POP Server**. It does not provide for sending Email, which is assumed to be done via **SMTP** or some other method. **POP** is useful for computers without a permanent **Network** connection which therefore require a "post office" (the **POP Server**) to hold their Email until they can retrieve it. **POP** is older and less powerful than the **IMAP**, which is used for the same application area.

## Port

While **IP** makes it possible for two systems to exchange datagrams in a heterogeneous network environment, it does not support the identification of different processes on these systems. **TCP** and **UDP** as the transport **Protocols** of the **Internet** thus introduce the concept of a **Port**, which addresses a specific process on a system. **Port** numbers are 16-bit numbers, which are included in **TCP** or **UDP** packets and are used to address a specific process on a system.

## PostScript

**PostScript** is an interpreted, stack-based **Programming Language**. Its primary application is to

describe the appearance of text, graphics, and images on printed or displayed pages. A program in [PostScript](#) can communicate a document description from a composition system to a printing system in a device-independent way. [PostScript](#) is an unusually powerful printer language because it is a full programming language, rather than a series of low-level escape sequences. The latest version of [PostScript](#), version 3, fully integrates [PDF](#).

## PPP - Point to Point Protocol

[PPP](#) is designed for simple links which transport packets between two peers. These links provide full-duplex simultaneous bi-directional operation, and are assumed to deliver packets in order. Although [PPP](#) is not tied to a particular type of packets it transports, its most common use is the encapsulation of [IP](#) packages over [Modem](#) lines. Basically, [PPP](#) is similar to [SLIP](#), but it has the advantages of not being limited to one type of [Protocol](#) it can transport, a configuration negotiation phase at the start of a connection (for determining connection configuration parameters automatically), and the possibility to use standardized authentication procedures for automated login. The two authentication schemes supported by [PPP](#) are PAP and CHAP.

## Product

The [Product](#) is a base topic for topics describing a picece of software or hardware.

## Programming Language

A [Programming Language](#) is a base topic for topics describing a language for writing executable programs.

## Protocol

The [Protocol](#) is a base topic for topics describing a defined way of transferring data between distributed peers. Essentially, a [Protocol](#) defines an [Interface](#) for distributed scenarios. In a way, a [Protocol](#) can be regarded as the distributed equivalent of a [API](#), because it is used for the same purposes and defines the same things (the possible interactions between components, and the data that is exchanged while interacting).

## Proxy

A [Proxy](#) is a special kind of [Gateway](#), acting like a [Server](#) when being accessed by a [Client](#). However, instead of servicing a request from a [Client](#), the [Proxy](#) forwards the request to a [Server](#), waits for the [Server](#)'s response, and then sends it back to the [Client](#). A [Proxy](#) is often combined with other typical [Gateway](#) functionality, such as [Firewall](#) or Cache.

## PSTN - Public Switched Telephone Network

[PSTN](#) refers to the established international telephone system carrying voice data over circuit switched connections. Newer developments such as ISDN are completely digital, while [PSTN](#) traditionally works on an analog basis (even though [PSTN](#)-based networks also increasingly are implemented using digital technologies).

### Public-Key Cryptography - Asymmetric Cryptography

[Public-Key Cryptography](#) is a Cryptography method where different keys are used for encryption and decryption. This is a major advantage over [Secret-Key Cryptography](#) and makes [Public-Key Cryptography](#) ideally suited for scenarios where secure key exchange is hard or impossible. The disadvantage of [Public-Key Cryptography](#) is that all known algorithms are very computing-intensive.

### Python

[Python](#) is an interpreted, interactive, object-oriented [Programming Language](#). The language has a concise syntax; a small number of powerful high-level data types are built in. [Python](#) can be extended in a systematic fashion by adding new modules implemented in a compiled [Programming Language](#) such as C or C++. Such extension modules can define new functions and variables as well as new object types. [Python](#) is frequently used for programming [CGI](#) applications.

### Query Language

A [Query Language](#) is a base topic for topics describing a language for formulating queries for structured data.

### QuickTime

[QuickTime](#) is Apple's architecture for handling multimedia data. First versions of [QuickTime](#) were basically a [Format](#) for audio and video, newer versions integrate streaming capabilities as well as more media types, including support for 3D and virtual reality. Being a proprietary technology, [QuickTime](#) can be compared to Microsoft's [ASF](#).

### QXGA - Quantum XGA

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 2048 by 1536 pixels.

### RDF - Resource Description Framework

[RDF](#) is designed to provide an infrastructure supporting [Meta Data](#) across many [WWW](#)-based activities. [RDF](#) is the result of a number of [Meta Data](#) communities bringing together their needs to provide a robust and flexible architecture for supporting [Meta Data](#) on the [Internet](#) and the [WWW](#). Example applications include site maps, content ratings, stream channel definitions, search engine data collection, digital library collections, and distributed authoring. [RDF](#) allows different application communities to define the [Meta Data](#) property set that best serves the needs of each community. [RDF](#) provides a uniform and interoperable means to exchange the [Meta Data](#) between programs and across the [WWW](#). Furthermore, [RDF](#) provides a means for publishing both a human-readable and a machine-understandable definition of the property set itself. [RDF](#) uses [XML](#) as the transfer syntax in order to leverage other tools and code bases being built around [XML](#).

### Repeater

A [Repeater](#) is a device that connects network segments on the physical layer of the network. Related to [Repeaters](#) are [Bridges](#), [Routers](#), and [Gateways](#), which also connect network segments, but on different layers of the networking architecture.

### Resolution Protocol

The [Resolution Protocol](#) is a base topic for topics describing protocols for resolution purposes.

### RFC - Request For Comments

[RFCs](#) form a series of publications of networking technical documents, started in 1969 as part of the original ARPA wide-area networking ([ARPANET](#)) project. [RFCs](#) cover a wide range of topics, from early discussion of new research concepts to status memos about the [Internet](#). The [IAB](#) views the [RFC](#) publication process to be sufficiently important to warrant including the [RFC](#) editor in the [IAB](#) membership. The status of specifications on the [Internet](#) standards track is summarized periodically in a summary [RFC](#) entitled "Internet Official Protocol Standards". This [RFC](#) shows the level of maturity and other helpful information for each [Internet Protocol](#) or service specification. The "Internet Official Protocol Standards" [RFC](#) is the authoritative statement of the status of any particular [Internet](#) specification, and it is the "Publication of Record" with respect to [Internet](#) standardization.

### RMI - Remote Method Invocation

[RMI](#) is an RPC mechanism enabling [Java](#) programmers to create distributed applications, in which the methods of remote [Java](#) objects can be invoked from another [JVM](#), possibly on a different host. A [Java](#) program can make a call on a remote object once it obtains a reference to the remote object, either by looking up the remote object in the bootstrap naming service provided by [RMI](#), or by receiving the reference as an argument or a return value. A [Client](#) can call a remote object in a [Server](#), and that [Server](#) can also be a [Client](#) of other remote objects. [RMI](#) uses object serialization to marshal and unmarshal parameters and does not truncate types, supporting true object-oriented polymorphism.

### Router

A [Router](#) is a device that connects network segments on the network layer of the network. Related to [Routers](#) are [Repeaters](#), [Bridges](#), and [Gateways](#), which also connect network segments, but on different layers of the networking architecture.

### Routing Protocol

The [Routing Protocol](#) is a base topic for topics describing protocols for facilitating routing within computer networks.

### RSA - Rivest, Shamir, and Adleman

**RSA** is a **Public-Key Cryptography** system for both encryption and authentication (its name is derived from the surnames of the three inventors). For encryption, **RSA** is combined with a **Secret-Key Cryptography** system, such as **DES**, to encrypt a message by means of an **RSA** digital envelope. For authentication, **RSA** is usually combined with a Cryptographic Hash Function, such as **MD5**, to sign a message.

### RTCP - Real Time Control Protocol

**RTCP** is the control **Protocol** that works in conjunction with **RTP**. **RTCP** control packets are periodically transmitted by each participant in an **RTP** session to all other participants. Feedback of information to the application can be used to control performance and for diagnostic purposes.

### RTP - Real Time Protocol

**RTP** provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video, or simulation data, over multicast or unicast network services. **RTP** does not address resource reservation and does not guarantee quality-of-service for real-time services. The data transport is augmented by a control protocol (**RTCP**) to allow monitoring of the data delivery in a manner scalable to large multicast networks, and to provide minimal control and identification functionality. **RTP** and **RTCP** are designed to be independent of the underlying transport and network layers.

### RTSP - Real Time Streaming Protocol

**RTSP** is an application-level **Protocol** for control over the delivery of data with real-time properties. **RTSP** provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video. Sources of data can include both live data feeds and stored clips. **RTSP** is intended to control multiple data delivery sessions, provide a means for choosing delivery channels such as **UDP**, multicast **UDP** and **TCP**, and use delivery mechanisms based upon **RTP**.

### SAX - Simple API for XML

**SAX** is a standard **API** for event-based **XML** parsing, and **SAX** implementations are available in different programming languages. **SAX** is the event-based alternative to the tree-based **DOM**, which also provides an **API** for accessing **XML** documents. An event-based **API** reports parsing events (such as the start and end of elements) directly to the application through callbacks, and does not usually build an internal tree. The application implements handlers to deal with the different events. The advantage of this approach is simpler processing, in particular the ability to process **XML** documents without the need to keep them in memory, which makes it possible to process **XML** documents that exceed the capacity of memory available for processing.

### Search Engine

The **Search Engine** is a base topic for topics describing programs which are crawling the **WWW** for collecting and indexing resources. A **Search Engine** typically interprets **HTML**

pages, but an increasing number of [Search Engines](#) also interprets other resources, such as [PDF](#) documents.

### Secret-Key Cryptography - Symmetric Cryptography

[Secret-Key Cryptography](#) is a Cryptography method where the same key is used for encryption and decryption. This leaves applications with the problem of how to securely exchange this key. This inherent problem of [Secret-Key Cryptography](#) has been solved by [Public-Key Cryptography](#) methods.

### Semiconductor Storage Medium

The [Semiconductor Storage Medium](#) is a base topic for topics describing media for semiconductor data storage.

### Server

The [Server](#) is a base topic for topics describing servers. A [Server](#) is one component in the [Client/Server-Model](#) and is contacted by a [Client](#) for providing a service of some kind.

### Server Side Technology

The [Server Side Technology](#) is a base topic for topics describing a [Technology](#) which is used on the [Server](#) side of a scenario based on the [Client/Server-Model](#).

### SET - Secure Electronic Transaction

[SET](#) is an industry-wide [Protocol](#) designed to safely transmit sensitive personal and financial information over public [Networks](#). [SET](#) contains state-of-the-art cryptographic technology that provides online transaction security that is equivalent or superior to the safeguards in present physical, Email, and telephone card transactions. [PKCS](#) is the set of [Public-Key Cryptography](#) algorithms used in [SET](#). The [Secret-Key Cryptography](#) algorithm is [DES](#).

### SGML - Standard Generalized Markup Language

[SGML](#) is a markup language for structured documents. Being the foundation for [HTML](#), [SGML](#) today is the most frequently used language for structuring documents. The rules for how documents of a certain type may be structured are specified in a [DTD](#), and every application of [SGML](#) (such as [HTML](#)) defines such a [DTD](#). Even though [SGML](#) has been very successful, it is also rather complex and contains a lot of obscure features which are rarely used (and implemented). Thus, when a new language for replacing [HTML](#) on the [WWW](#) was needed, rather than directly taking [SGML](#), a functional subset of [SGML](#) was defined, which has become known under the name of [XML](#).

### SHOE - Simple HTML Ontology Extensions

[SHOE](#) is an [XML](#)-based knowledge representation language, a superset of [HTML](#) which adds the tags necessary to embed arbitrary semantic data into [WWW](#) pages. [SHOE](#) tags are divided

into two categories. First, there are tags for constructing ontologies. **SHOE** ontologies are sets of rules which define what kinds of assertions **SHOE** documents can make and what these assertions mean. Secondly, there are tags for annotating **SHOE** documents to subscribe to one or more ontologies, declare data entities, and make assertions about those entities under the rules prescribed by the ontologies.

## S-HTTP - Secure Hypertext Transfer Protocol

**S-HTTP** is an extension of **HTTP** providing independently applicable security services for transaction confidentiality, authenticity/integrity and non-repudiability of origin. The protocol emphasizes maximum flexibility in choice of key management mechanisms, security policies and cryptographic algorithms by supporting option negotiation between parties for each transaction. Message protection can be provided on three orthogonal axes: signature, authentication, and encryption. Any message may be signed, authenticated, encrypted, or any combination of these (including no protection). Several cryptographic message format standards may be incorporated into **S-HTTP** clients and servers, particularly, but in principle not limited to, PKCS-7 and PEM. **S-HTTP**-aware clients can communicate with **S-HTTP**-oblivious servers and vice-versa. Cryptographic algorithms supported by **S-HTTP** include **DES**, two-key and three-key 3DES, DESX, **IDEA**, RC2, and CDMF.

## SLIP - Serial Line Internet Protocol

**SLIP** is a packet framing **Protocol**, it defines a sequence of characters that frame **IP** packets on a serial line. It provides no addressing, packet type identification, error detection/correction, or compression mechanisms. It is used for the same purpose as **PPP**, which is the encapsulation of **IP** packages over **Modem** lines. **SLIP** does not have **PPP**'s configuration negotiation or authentication schemes, which can make the configuration of **SLIP** connections more complicated.

## SMS - Short Message Service

The **GSM SMS** is a very simple service for transmitting short messages (140 octets of user data) over the **GSM** network. It has become a very popular service for mobile phone users. The EMS and MMS technologies have been designed to become the predecessors of **SMS**.

## SMTP - Simple Mail Transfer Protocol

**SMTP** is used to pass Email messages between **Internet Servers**. Each message has a standardized header that is used to identify Email address(es) of the person(s) the message is to be sent to, the Email address and name of the sender (to whom responses can be sent automatically), and details of those nodes on the network through which the message passed. A number of extensions to **SMTP** have been defined yielding **ESMTP**, which is mostly in use today.

## SPEC - Standard Performance Evaluation Corporation

**SPEC** was founded in 1988 by a small number of workstation vendors. **SPEC** has grown to become one of the more successful performance standardization bodies with more than 40 member companies. **SPEC** publishes several hundred different performance results each

quarter spanning across a variety of system performance disciplines. The goal of [SPEC](#) is to ensure that the marketplace has a fair and useful set of metrics to differentiate candidate systems. The path chosen is an attempt to balance between requiring strict compliance and allowing vendors to demonstrate their advantages.

## SPECweb

[SPECweb](#) a standardized benchmark for comparing [HTTP Server](#) performance. The benchmark is designed to provide comparable measures of how well systems can handle [HTTP GET](#) requests. The workload is based upon analysis of [HTTP Server](#) logs from [WWW](#) sites ranging from a small personal server up through some of the [WWW](#)'s most popular servers.

## SPIFF - Still Picture Interchange File Format

[SPIFF](#) is the "official" [Image Format](#) for images using the [JPEG Image Compression](#) algorithm. Part 3 of the [JPEG](#) standard includes a fully defined [Image Format](#) for storing [JPEG](#) data. When the [JPEG](#) format was first standardized, disagreements among ISO committees prevented a standard [JPEG Image Format](#) from being created. The de-facto format that appeared was [JFIF](#) from C-cube Microsystems. The [JFIF Image Format](#), although now widespread, is very limited in capability as [Image Formats](#) go. [SPIFF](#) is intended to replace [JFIF](#), adding features (more color spaces, a recognized way of including text blocks, and so forth), and providing a backwards-compatibility allowing [SPIFF](#) files to be read by most [JPEG/JFIF](#) decoders. [JFIF](#), however, has a five-year head start on [SPIFF](#), so the likelihood of it being completely replaced anytime soon is not good.

## sRGB - standard RGB

The aim of the [sRGB Color Space](#) is to complement the current color management strategies by enabling a third method of handling color in [Operating Systems](#), device drivers and the [Internet](#) that utilizes a simple and robust device independent color definition. This will provide good quality and backward compatibility with minimum transmission and system overhead. Based on a calibrated colorimetric RGB color space well suited to CRT monitors, television, scanners, digital cameras, and printing systems, such a space can be supported with minimum cost to software and hardware vendors.

## SSI - Server-Side Includes

[SSI](#) make it possible to include information into [WWW](#) pages before delivering them to a [HTTP Client](#). A [WWW](#) page using [SSI](#) contains special instructions which are interpreted by the [HTTP Server](#) whenever the [WWW](#) page is requested. These instructions may specify to include other documents (eg, document headers or footers) or to insert dynamic information, such as the current date or an access count. There is no standard for [SSI](#), so each [HTTP Server](#) implementation uses its own syntax and functionality.

## SSL - Secure Sockets Layer

The primary goal of [SSL](#) is to provide privacy and reliability between two communicating applications. [SSL](#) is composed of two layers. At the lower level, layered on top of some reliable

transport [Protocol](#), for example [TCP](#), is the [SSL Record Protocol](#), which is used for encapsulation of various higher level protocols. One such encapsulated protocol, the [SSL Handshake Protocol](#), allows the server and client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys before the application protocol transmits or receives its first byte of data. One advantage of [SSL](#) is that it is application [Protocol](#) independent. A higher level [Protocol](#) can layer on top of [SSL](#) transparently. For [Internet](#) applications, a generalized variant of [SSL](#) called [TLS](#) has been developed.

### Storage Medium

The [Storage Medium](#) is a base topic for topics describing media for data storage.

### Style Sheet Language

The [Style Sheet Language](#) is a base topic for topics describing a way of specifying style sheets for documents. A style sheet is used to control the presentation of a document. A [Style Sheet Language](#) can be regarded as [Meta Data](#) because it specifies the presentation aspects of an information resource, hence it specifies data about data.

### SVG - Scalable Vector Graphics

[SVG](#) is a language for describing two-dimensional graphics in [XML](#). [SVG](#) allows for three types of graphic objects: vector graphic shapes (eg, paths consisting of straight lines and curves), images, and text. Graphical objects can be grouped, styled, transformed and composited into previously rendered objects. Text can be in any of the [XML Namespaces](#) suitable to the application, which enhances searchability and accessibility of the [SVG](#) graphics. The feature set includes nested transformations, clipping paths, alpha masks, filter effects, template objects and extensibility. [SVG](#) drawings can be dynamic and interactive. [DOM](#) for [SVG](#), which includes the full [XML DOM](#), allows for straightforward and efficient vector graphics animation via scripting. Event handlers can be assigned to any [SVG](#) graphical object. Because of its compatibility and leveraging of other [WWW](#) standards, features like scripting can be done on [SVG](#) elements and other [XML](#) elements from different [XML Namespaces](#) simultaneously within the same [WWW](#) page.

### SVGA - Super VGA

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 800 by 600 pixels.

### SXGA - Super XGA

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 1280 by 1024 pixels.

### SXGA+ - Super XGA+

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 1400 by 1050 pixels.

## T9 - Text on 9 keys

**T9** is a technology for rapidly typing alphanumeric text on a numeric keypad. It is based on a dictionary and thus language-dependent. **T9** is often used on mobile phones to facilitate text input for messaging, for example for composing **SMS** messages.

## Tcl - Tool Command Language

**Tcl** is a general-purpose, robust **Programming Language** that can easily be integrated into new applications. One of **Tcl**'s most useful features is its extensibility. If an application requires some functionality not offered by standard **Tcl**, new **Tcl** commands can be implemented using the C language, and integrated fairly easily. Since **Tcl** is so easy to extend, many people have written extension packages for common tasks, and made these freely available. **Tcl** is frequently used for programming **CGI** applications.

## TCP - Transmission Control Protocol

**TCP** is intended for use as a highly reliable transport **Protocol** between hosts in packet-switched computer communication **Networks**, and in interconnected systems of such **Networks**. **TCP** is a flow-controlled, connection-oriented, end-to-end reliable **Protocol** designed to fit into a layered hierarchy of **Protocols** supporting multi-network applications. **TCP** provides for reliable interprocess communications between pairs of processes in host computers attached to distinct but interconnected computer communication **Networks**. Very few assumptions are made as to the reliability of the communication **Protocols** below the **TCP** layer. **TCP** assumes it can obtain a simple, potentially unreliable, datagram service from the lower level **Protocols**, usually **IP**. **TCP** is able to operate above a wide spectrum of communication systems, ranging from hard-wired connections to packet-switched or circuit-switched networks.

## Technology

The **Technology** is a base topic for topics describing an applicable technological achievement.

## Telephony Network

The **Telephony Network** is a base topic for topics describing networks for telephony.

## Telnet

The purpose of the **Telnet Protocol** is to provide a fairly general, bi-directional, 8-bit byte oriented communications facility. Its primary goal is to allow a standard method of interfacing terminal devices and terminal-oriented processes to each other. The most popular usage of **Telnet** is for logging in into remote systems. In this scenario, the **Telnet Client** is the remote terminal (usually running some kind of terminal emulation) which is connected to a terminal driver program using the **Telnet Protocol**.

## TeX

**TeX** is a typesetting system for high-quality document preparation. **TeX** itself does not provide a **GUI** like most word processing programs, but is invoked to process a document which contains commands and the actual content. Because of its superior formatting of mathematical formulae, **TeX** is very popular in the scientific community. **TeX** is most often used in the form of **LaTeX**, which is a macro package based on **TeX**. Another popular tool in the context of **TeX** is **BibTeX**, a program for processing bibliographic databases.

### Text Format

The **Text Format** is a base topic for topics describing a defined way of coding information for storage or transfer of text.

### THX

**THX** defines a set of guidelines for audio and picture playback for movie theatres. Often confused with **Movie Sound Formats**, **THX** does not define standards for audio coding, it only defines the presentation standards that must be met during movie playback. Any **Movie Sound Format** may be used, as long as it provides the standards set by **THX**.

### TIFF - Tag Image File Format

**TIFF** defines a tag-based **Image Format** that can characterize almost any form of 2D raster data using either **ASCII** or binary coding. "Private" tags may be used to allow additional parameters to be added to the descriptor. "Standard" **TIFF** allows the use of PackBits, **LZW**, G3 Fax, G4 Fax, and **JPEG** compression schemes within transmitted images. Four photometric classes are supported: TIFF-B for monochrome, TIFF-G for grayscale, TIFF-P for palette-based coding, and TIFF-R for RGB coding.

### TLD - Top-Level Domain

A **TLD** is that part of a **DNS FQDN** which stands right of the rightmost full stop. Two letter **TLDs** designate **ccTLDs**, and three letter **TLDs** designate **gTLDs**.

### TLS - Transport Layer Security

**TLS** was developed as the successor to **SSL**, and is nearly identical to **SSL**, except that it implements an open and standards-based solution, more non-proprietary ciphers, better error reporting, and HMAC digests instead of simple **MD5**. The structure of the start of a **TLS** session allows negotiation of the level of the protocol to be used. This way, a **Client** or **Server** can simultaneously support **TLS** and **SSL** and negotiate the most appropriate protocol for the connection.

### topic

This is the base object from which (directly or indirectly) all topics must be derived. in a perfect system, this object would probably be built-in, but this isn't a perfect system...

### Topic Maps

[Topic Maps](#) enable vast information resources (such as the [WWW](#)) to be classified and navigated in a consistent manner. They allow for the concepts or topics that underlie a set of information resources to be exposed to those people or applications processing the information. Information resources can be [HTML](#), [PDF](#), [XML](#), [SGML](#) and other [Formats](#) including paper. Topics are the concepts underlying what these resources are concerned with. A topic can reference zero or more information resources. In addition to grouping resources as "topic occurrences", it is possible to have meta level associations. This means that it is possible to define a link, or association, from one topic to another. [Topic Maps](#) provide a semantic layer that is not hierarchical, although it could be visualized that way, it facilitates navigation at a level independent from the information resources.

## TrAX - Transformation API for XML

[TrAX](#) is an [API](#) for transforming [XML](#) documents using [XSLT](#) style sheets. [TrAX](#) is a [Java API](#) and has been defined to provide common access to different [XSLT Processors](#). [TrAX](#) is part of the [JAXP API](#), which combines a number of [Java APIs](#).

## TrueType

[TrueType](#) is a [Font Format](#) developed by Apple and licensed to Microsoft. [TrueType](#) is the native [Operating System Font Format](#) for Windows and MacOS. [TrueType](#) contains a hierarchical set of tables and glyph representations. Characters can be hinted on a per character and point size basis yielding excellent quality at screen resolutions. [TrueType](#) fonts for Windows and MacOS have few differences, though they can be different enough to prevent cross platform usage. Font foundries provide [TrueType](#) fonts for each platform and usually include a license preventing electronic manipulation to achieve cross platform transparency. [TrueType](#) is one of the foundations for the [OpenType Font Format](#).

## Type 1

Originally developed by Adobe for their [PostScript](#) page description language, the [Type 1 Font Format](#) has been accepted as an ISO standard. [Type 1](#) fonts use a specialized subset of the [PostScript](#) language which is optimized for better performance and a more compact representation. The [Type 1](#) operator set includes hint information which helps font rasterizers create more accurate bitmaps for smaller sizes and lower resolutions. [Type 1](#) is one of the foundations for the [OpenType Font Format](#).

## UCS - Universal Multiple-Octet Coded Character Set

[UCS](#) standardized in ISO 10646 integrates all previous internationally/nationally agreed character sets into a single code set. [UCS](#) is based on 4-octet (32-bit) coding scheme known as the "canonical form" (UCS-4), but a 2-octet (16-bit) form (UCS-2) is used for the BMP, where octets 1 and 2 are assumed to be 00 00. The code set is split into 128 "groups" of "planes" containing 256 "rows" with 256 "cells" for characters. Each character is addressed using multiple octets, the third (in UCS-2 the first) of which identifies the row containing the character and the fourth (in UCS-2 the second) its cell number. The first 127 characters of the BMP used for 16-bit code interchange are those of [ASCII](#). The characters forming the second half of the first row are those used in ISO 8859-1, the Latin-1 character set.

## UDDI - Universal Description, Discovery, and Integration

**UDDI** provides a standardized method for publishing and discovering information about Web services. **UDDI** is an industry initiative that attempts to create a platform-independent, open framework for describing services, discovering businesses, and integrating business services. **UDDI** focuses on the process of discovery in the service-oriented architecture (WSDL is often used for service description, and SOAP for service invocation). Being a Web service itself, **UDDI** is invoked using SOAP. In addition, **UDDI** also defines how to operate servers and how to manage replication among several servers.

## UDP - User Datagram Protocol

**UDP** is a transport **Protocol** that provides a simple but unreliable datagram service. **UDP** neither guarantees delivery nor does it require a connection. As a result, it is lightweight and efficient, but all error processing and retransmission must be taken care of by the application program. Like **TCP**, which is the **Internet**'s other important transport **Protocol**, **UDP** is layered on top of **IP**.

## UNC - Universal Naming Convention

**UNC** provides a naming convention for identifying network resources. **UNC** identifiers consist of three parts, a server name, a share name, and an optional file path, that are combined using backslashes or slashes. **UNC** notation is used primarily for mapping network drives in the Windows family of **Operating Systems**, although support for **UNC** appears in related technologies like SMB/CIFS. **UNC** names are most commonly used to reach file servers or printers on a LAN.

## Unicode

**Unicode** defines a 31-bit character set. **Unicode** is closely aligned with **UCS**. The most commonly used characters, including all those found in older encoding standards, have been placed in one of the first 65534 positions (0x0000 to 0xFFFFD). This 16-bit subset is called the BMP or "Plane 0". The characters that were later added outside the 16-bit BMP are mostly for specialist applications such as historic scripts and scientific notation. New characters are still being added on a continuous basis, but the existing characters will not be changed any more and are stable. **Unicode** assigns to each character not only a code number but also an official name. A hexadecimal number that represents a **Unicode** or **UCS** value is commonly preceded by "U+" as in U+0041 for the character "Latin capital letter A". The **Unicode** characters U+0000 to U+007F are identical to those in **ASCII**, and the range U+0000 to U+00FF is identical to ISO 8859-1 (Latin-1).

## URI - Uniform Resource Identifier

The **WWW** is considered to include objects accessed using an extendable number of **Protocols**, existing, invented for the **WWW** itself, or to be invented in the future. Access instructions for an individual object under a given **Protocol** are encoded into forms of address string. Other **Protocols** allow the use of object names of various forms. In order to abstract the idea of a generic object, the **WWW** needs the concepts of the universal set of objects, and of

the universal set of names or addresses of objects. A [URI](#) is a member of this universal set of names in registered name spaces and addresses referring to registered [Protocols](#) or name spaces. A [URL](#) is a form of [URI](#) which expresses an address mapping onto an access algorithm using network [Protocols](#). A [URN](#) is a form of [URI](#) which uses a name space (and associated [Resolution Protocols](#)) for persistent object names.

## URL - Uniform Resource Locator

A [URL](#) is the address of a resource which is retrievable using [Protocols](#) already deployed on the [Internet](#). A [URL](#) defines an access [Protocol](#), called a "scheme", and a "scheme-dependent part", which has to provide sufficient information to locate an object using the specified scheme. In case of [HTTP URLs](#), the scheme is "http", and the scheme-dependent part specifies the name of the [HTTP Server](#) as well as the path of the object on the [HTTP Server](#).

## URN - Uniform Resource Name

A [URN](#) is a persistent, globally unique name assigned to an object. In contrast to a [URL](#), which changes whenever the location of an object changes, a [URN](#) has no location dependence and therefore a longer lifetime. This is realized by using a naming service which in most cases will provide a mapping from URNs to URLs. Thus, even if the [URL](#) of an object changes, its [URN](#) remains the same, since only the object's entry in the naming service has to be updated.

## Usenet News - News

[Usenet News](#) can be regarded as the oldest [Internet](#) application for global distribution of information. It originated in 1979 at the University of North Carolina. It is simply a set of machines ([Usenet News Servers](#)) which cooperate to exchange (using [NNTP](#)) articles tagged with one or more labels, which are called "newsgroups". Newsgroups are organized hierarchically according to the subjects of the articles belonging to them. Newsgroups or individual articles in newsgroups can be addressed using a special [URL](#) scheme.

## UTF - UCS Transformation Format

[UTF](#) is used for coding [UCS](#) characters. Although [UCS](#) defines character codings (UCS-2 and UCS-4), they are hard to use in many current applications and protocols that assume 8- or even 7-bit characters. UTF formats usually are variable length formats, for example in UTF-8 a character is represented by 1 to 6 bytes, while in UTF-16 a character is represented by 2 to 4 bytes ([UTF-32](#), however, always encodes characters as 4 bytes).

## UTF-32

[UTF-32](#) is the [UTF](#) that serializes a [Unicode](#) code point as a sequence of four bytes, in either big-endian (UTF-32BE) or little-endian UTF-32LE format. An initial sequence corresponding to U+FEFF is interpreted as a BOM, it is used to distinguish between the two byte orders. The BOM is not considered part of the content of the text. A serialization of [Unicode](#) code points into [UTF-32](#) may or may not begin with a BOM. The term [UTF-32](#) can be used ambiguously. When referring to the encoding of [Unicode](#) in memory, there is no associated byte orientation,

and a BOM is never used. When referring to a serialization of [Unicode](#) code points into bytes, it may have a BOM and either byte orientation..

### UXGA - Ultra XGA

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 1600 by 1200 pixels.

### VBScript - Visual Basic Scripting Edition

Microsoft's [VBScript](#) is a subset of the Microsoft Visual Basic [Programming Language](#). [VBScript](#) is a portable, lightweight interpreter for use in [Browsers](#) and other applications that use [ActiveX](#) controls.

### VGA - Video Graphics Array

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 640 by 480 pixels.

### Video Compression

The [Video Compression](#) is a base topic for topics describing a defined way of compressing video streams.

### Video Format

The [Video Format](#) is a base topic for topics describing a defined way of coding information for storage or transfer of moving images.

### VPN - Virtual Private Network

A [VPN](#) is a private [Network](#) that makes use of the public telecommunication infrastructure, maintaining privacy through the use of a tunneling [Protocol](#) and security procedures. A [VPN](#) can be contrasted with a system of owned or leased lines that can only be used by one company. The main purpose of a [VPN](#) is to give the company the same capabilities as private leased lines at much lower cost by using the shared public infrastructure.

### VRML - Virtual Reality Modeling Language

[VRML](#) is a [Format](#) for 3D multimedia and shared virtual worlds on the [WWW](#). In comparison to [HTML](#), [VRML](#) adds the next level of interaction, structured graphics, and extra dimensions (z and time) to the presentation of documents. The applications of [VRML](#) are broad, ranging from simple business graphics to entertaining [WWW](#) page graphics, manufacturing, scientific, entertainment, and educational applications, and 3D shared virtual worlds and communities.

### W3C - World Wide Web Consortium

Founded in 1994 to develop common [Protocols](#) for the evolution of the [WWW](#), [W3C](#) is an in-

ternational association of industrial and service companies, research laboratories, educational institutions, and organizations of all sizes. All of these organizations share a compelling interest in the long term evolution and stability of the [WWW](#). [W3C](#) is a non-profit organization funded partly by commercial members. Its activities remain vendor neutral, however. [W3C](#) also receives the support of governments who consider the [WWW](#) the platform of choice for a global information infrastructure.

### WAE - Wireless Application Environment

[WAE](#) is the uppermost layer in the [WAP](#) software stack, this layer provides basic components on which [ASPs](#) can develop their mobile applications.

### WAIS - Wide Area Information Servers

[WAIS](#) is designed to help users find information over a computer network. The [WAIS](#) software architecture has four main components: the [Client](#), the [Server](#), the database, and the [Protocol](#). The [WAIS Client](#) is a UI that sends requests for information to local or remote [Servers](#). The [WAIS Server](#) is a program that services [Client](#) requests. The [Server](#) generally runs on a machine containing one or more information sources, or [WAIS](#) databases. The [Protocol](#), [Z39.50-1988](#), is used to connect [WAIS Clients](#) and [Servers](#) and is based on the 1988 version of the NISO [Z39.50](#) standard. Since the [WWW](#) allows greater flexibility in the structure and presentation of distributed information, the usage of [WAIS](#) services and the number of [WAIS Servers](#) is getting smaller.

### WAP - Wireless Application Protocol

[WAP](#) enables mobile users with wireless devices to easily access and interact with information services. [WAP](#) is not a single standard for data communications or document formats, but a whole suite of specifications ranging from data transport [Protocols](#) to the [WML](#), which is a markup language similar to [HTML](#), but based on [XML](#) and specifically designed for wireless devices. There are five layers to the [WAP](#) stack: the application layer ([WAE](#)), the session layer ([WSP](#)), the transaction layer ([WTP](#)), the security layer ([WTLS](#)), and the transport layer ([WDP](#)).

### WAVE - Waveform Audio File Format

[WAVE](#) is a proprietary standard for audio files developed by Microsoft. The format can store monaural or multichannel sampled sounds in a range of sampling types, sample rates and sample resolutions.

### WBXML - Wireless Binary XML

[WBXML](#) is a binary format for [WML](#), the [Hypertext Format](#) of [WAP](#). [WBXML](#) may either be delivered by the origin server itself, or the origin server delivers [WML](#), and the [WAP Gateway](#) performs the transformation between [WML](#) and [WBXML](#). [WBXML](#) works by defining a tokenization method for [WML](#), where each element and each attribute is represented by a compact binary token.

### WebCGM

**WebCGM** is a profile of the ISO **CGM** standard, tailored to the requirements for scalable 2D vector graphics in electronic documents on the **WWW**. The **WebCGM** profile is a subset of the **CGM** standard, and a set of specifications targeted especially at the effective application of the **CGM** standard to representation of 2D graphical content within **WWW** documents.

### WebDAV - WWW Distributed Authoring and Versioning

**WebDAV** defines extensions to **HTTP** to enable distributed **WWW** authoring tools to be broadly interoperable. **HTTP** already contains functionality which enables the editing of **WWW** content at a remote location, without direct access to the storage media via an **Operating System**. This capability is exploited by several existing **HTML** distributed authoring tools, and by a growing number of mainstream applications (eg, word processors) which allow users to write (publish) their work to an **HTTP Server**. Experience from the **HTML** authoring tools has shown they are unable to meet their user's needs using the facilities of **HTTP**. The consequence of this is either postponed introduction of distributed authoring capability, or the addition of nonstandard extensions to **HTTP**. These extensions, developed in isolation, are not interoperable.

### WebFonts

The **WebFonts** specification is part of CSS2, it allows improved client-side font matching, enables font synthesis and progressive rendering, and enables fonts to be downloaded over the **WWW**. However, **WebFonts** does not specify a particular data format for downloadable fonts, it only defines a mechanism for font description.

### WebSGML

The **SGML** standard has been updated with two annexes which add some corrections as well as new features to **SGML** making some specifications possible which are desirable for using **SGML** as the basis for **HTML** and **XML**. Basically, the **WebSGML** extensions allow a number of additional features to be defined in an **SGML** declaration and **DTD**. However, when using these features in an **SGML** environment, it is necessary that both the generator and the interpreter of a document are capable of processing the **WebSGML** extensions, since a conforming **SGML** implementation does not have to implement the **WebSGML** extensions.

### WebStone

**WebStone** is a benchmark for measuring the performance of **HTTP Server** platforms (software and hardware combined). It is designed to measure the performance of **HTTP Servers** under multiple scenarios which reflect different **WWW** site profiles. The test uses workload parameters and clients to generate **HTTP** traffic that allows an **HTTP Server** to be stressed in a number of different ways.

### Wi-Fi - Wireless Fidelity

**Wi-Fi** is a label for devices conforming to the IEEE 802.11b standard for **WLAN**. The IEEE

802.11b standard has been published by the IEEE, which does not perform conformance testing. In order to establish such a conformance testing process, the WECA has been formed, which tests devices for conformance with the IEEE 802.11b standard and issues the [Wi-Fi](#) label for conforming devices.

### Wireless Interface

The [Wireless Interface](#) is a base topic for topics describing ways for transmitting data between devices using wireless techniques.

### WLAN - Wireless LAN

A [WLAN](#) is a LAN that is based on a [Wireless Interface](#). [WLAN](#) technology makes using LANs easier, because users can roam within the coverage area of the [WLAN](#) without having to deal with cables.

### WML - Wireless Markup Language

[WML](#) is the [Hypertext Format](#) for [WAP](#). [WML](#) is based on [XML](#) (it is formally defined by an [XML DTD](#)). [WML](#) pages can be stored on any [HTTP Server](#), which will be contacted by a [WAP Gateway](#) and deliver the [WML](#) page. The [WAP Gateway](#) then transforms the [WML](#) page into [WBXML](#), which is then transmitted to the [WAP](#) device. [WML](#) is similar to [HTML](#) in its design, and [WAP](#) also defines [WMLScript](#) which may be used to implement dynamic [WML](#).

### WSP - Wireless Session Protocol

[WSP](#) forms the interface between [WAE](#) and the rest of the [WAP](#) stack. [WSP](#) operates in two modes, connectionless and connection, supports the transmission of [WML](#), and can be regarded as the binary equivalent of [HTTP](#).

### WSXGA+ - Wide SXGA+

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 1680 by 1050 pixels.

### WTA - Wireless Telephony Application

[WTA](#) is a part of [WAP](#)'s [WAE](#). [WTA](#) is an application framework for telephony services. [WTA](#) provides four services: [WTAI](#), repository, event handling, and [WTA](#) service information.

### WTAI - Wireless Telephony Application Interface

[WTAI](#) defines a set of [WTA](#)-related functions in a wireless device that can be invoked via [WML](#) or [WMLScript](#).

### WTLS - Wireless Transport Layer Security Protocol

**WTLS** sits between WTP and WDP layers in the **WAP** stack, and is responsible for providing transport layer security between the **WAP** client and **WAP Gateway** or **Proxy**. **WTLS** is optimized for narrowband communication.

### WUXGA - Wide UXGA

A **Display Standard** referring to a video adapter capable of a resolution of up to 1920 by 1200 pixels.

### WWW - World Wide Web

The **WWW** is a distributed **Hypermedia System** which is built on top of some of the services provided by the **Internet**. It is based on the **Client/Server-Model**, defining a **Protocol** for the exchange of information between **WWW Clients** and **Servers**.

### X11 - X Window System

**X11** is a distributed window system that is based on the **Client/Server-Model**. In **X11**, the traditional roles of a **Client** and a **Server** are not distributed as usual, because the **X11 Server** is the local entity (providing the service of displaying data and accepting user input), while the **X11 Client** is the remotely running program, which uses the **X11 Server** to provide a **GUI** to the user. **X11** is based on a **Protocol**, which makes it very easy to separate a (possibly heavy-weight) program and its **GUI**. **X11** is commonly used in Unix-derived **Operating Systems**.

### X3D - Extensible 3D

**X3D** is the name under which the development of **VRML** is continued. **X3D** is based on **XML** and is backwards compatible with **VRML**. Furthermore, it is componentized, profiled, and extensible, which makes it possible to use **X3D** in very different scenarios, from high-end visualizations to lightweight applications.

### X.500

**X.500** is an open, distributed, online directory service which is intended to be global in scope. **X.500** is a support service for data exchange which includes providing directory support for data communication services specified by other **OSI** application standards. The **X.500** series of standards covers services available to users, the functional model and protocols connecting the component parts of the directory, an information framework and a schema of the information held by the directory, and a mechanism for allowing **OSI** components to authenticate each other.

### X.509 - Directory Authentication Framework

**X.509** describes two levels of authentication, simple authentication, based on use of a password to verify user identity, and strong authentication, using credentials created by cryptographic methods. The standard recommends that only strong authentication should be used as the basis of providing secure services. **Public-Key Cryptography** is used for strong authentication, but **X.509** is not dependent on the use of a particular cryptographic algorithm, though

two users wishing to authenticate must support the same algorithm.

## Xanadu

[Xanadu](#) is an overall paradigm, an ideal and general model for all computer use, based on sideways connections among documents and files. This paradigm is especially concerned with electronic publishing, but also extends to all forms of storing, presenting and working with information. It is a unifying system of order for all information, non-hierarchical and side-linking, including electronic publishing, personal work, organization of files, corporate work and groupware. All data (for instance, paragraphs of a text document) may be connected sideways and out of sequence to other data (for instance, paragraphs of another text document). This requires new forms of storage, and invites new forms of presentation to show these connections. On a small scale, the paradigm means a model of word processing where comments, outlines and other notes may be stored conceptually adjacent to a document, linked to it sideways. On a large scale, the paradigm means a model of publishing where anyone may quote from and publish links to any already-published document, and any reader may follow these links to and from the document.

## XCMS - XML Cryptographic Message Syntax

[CMS](#) is used to digitally sign or encrypt arbitrary messages. [CMS](#) describes an encapsulation syntax for data protection. It supports digital signatures and encryption. The syntax allows multiple encapsulation, so one encapsulation envelope can be nested inside another. Likewise, one party can digitally sign some previously encapsulated data. It also allows arbitrary attributes, such as signing time, to be authenticated along with the message content, and provides for other attributes such as counter-signatures to be associated with a signature.

## XDuce

[XDuce](#) is a typed [Programming Language](#) that is specifically designed for processing [XML](#). One can read an [XML](#) document as an [XDuce](#) value, extract information from it or convert it to another format, and write out the result value as an [XML](#) document. Since [XDuce](#) is statically typed, [XDuce](#) programs never yield run-time type errors and the resulting [XML](#) documents always conform to specified types.

## XER - XML Encoding Rules

[ASN.1 XER](#) is a set of encoding rules that encode [ASN.1](#) data in [XML](#) document syntax. While this encoding is not as compact as other encoding rules commonly used for [ASN.1](#), it enables users to on the one hand use [ASN.1](#)'s powerful syntax, and on the other hand generate data from it which can be processed using existing [XML](#) tools (such as [XSLT](#)). Furthermore, because [XER](#) uses [XML](#) syntax, it is human-readable, while all other [ASN.1](#) encoding rules generate binary data.

## XGA - Extended Graphics Array

A [Display Standard](#) referring to a video adapter capable of a resolution of up to 1024 by 768 pixels.

### XHTML - Extensible Hypertext Markup Language

**XHTML** is a reformulation of **HTML** in **XML**. **HTML** is based on **SGML** and uses some features of **SGML** which are not available in **XML** (most notably, markup minimization), and consequently **HTML** documents typically are not valid **XML** documents. **XHTML** redefines **HTML** as an **XML DTD**, and also gives some recommendations how to use the markup in order to make **XHTML** compatible with older browsers, which only understand **HTML** (and thus may have problems processing **XML** syntax, for example the special form of empty elements).

### XLink - XML Linking Language

**XLink** defines how to insert links in **XML** documents. It specifies a framework making it possible for **XML** applications to recognize **XML** elements as having link semantics. In addition to the simple, two-ended, unidirectional links which are well-known from **HTML**, **XLink** allows more general links, which must not be embedded in the document, can have any number of ends, and can be multidirectional.

### XLL - Extensible Linking Language

**XLL** was the term under which the development of **XLink** and **XPointer** started, referring to both activities as a whole, but it is no longer used.

### XML - Extensible Markup Language

**XML** is a markup language for structuring arbitrary data. **XML** was designed to replace **HTML**, which was deemed too restricted with its fixed set of elements and attributes. Because **HTML** is based on **SGML**, but **SGML** itself was considered as being too complex for direct application on the **WWW**, **XML** was defined as a functional subset (a "profile") of **SGML**. **XML** defines data types (called "schemas") with **DTDs**, which originate from the document-centered view of **SGML**. However, **XML** is very successful in **B2B** scenarios, and as such is increasingly used for data exchange (as opposed to document exchange). Because data description requires other features than document description (eg, built-in data types and type derivation), **XML Schema** has been defined as a replacement for **DTDs**, which makes **XML** more usable for **B2B** scenarios.

### XML Infoset - XML Information Set

**XML Infoset** is an abstract data set describing the information available from an **XML** document. For many applications, this way of looking at an **XML** document is more useful than having to analyze and interpret **XML** syntax. **DOM** describes an **API** through which the information in an **XML Infoset** (ie, the information available from a specific **XML** document) can be accessed from different programming languages.

### XML Namespaces

**XML Namespaces** are used to qualify unique names in **XML Namespaces** documents which use schemas from different sources. This can occur because schemas (such as **DTDs**) are re-

used. However, if schemas are combined, it is possible that name conflicts appear. [XML Namespaces](#) defines a way how schema identification (through a [URI](#)) and names of a schema are combined to yield unique names.

### XML Processor - XML Parser

An [XML Processor](#) is a base topic for topics describing programs which are [XML](#) processors. These may be non-validating or validating [XML](#) processors (validating an [XML](#) document against its [DTD](#)), or they may support another [XML Schema Language](#), such as [XML Schema](#).

### XML Schema

[XML Schema](#) is an [XML Schema Language](#) for [XML](#). After [XML](#)'s success for [B2B](#) applications, it quickly became apparent that [XML](#)'s built-in [XML Schema Language](#), the [DTD](#), did not meet the requirements of application developers. [XML Schema](#) has thus been standardized by the [W3C](#) and extends [DTD](#) considerably, in particular with a type system supporting type derivation, and with a number of built-in simple types (such as integer, float, date, and time). While [XML Schema](#) is much more powerful than [DTDs](#), it is also much more complex, but it is very likely that over time [XML Schema](#) will replace [DTDs](#).

### XML Schema Language

An [XML Schema Language](#) is a method for specifying a grammar or rules for a class of [XML](#) documents. Even though the [XML](#) standard itself already contains such an [XML Schema Language](#) (the [DTD](#)), this [XML Schema Language](#) has a number of shortcomings, most notably no type derivation, and almost no support for data types. Consequently, a number of [XML Schema Languages](#) have been proposed, which aim at providing a more powerful language for specifying grammars for [XML](#) documents. [W3C](#)'s [XML Schema](#) is the most promising prospect for becoming the [XML Schema Language](#) to replace [DTDs](#).

### XML-Data

[XML-Data](#) is a proposal authored by Microsoft for the definition of [WWW Meta Data](#). The proposal has been used as input for [W3C](#)'s work on [RDF](#), which has the same application area as [XML-Data](#).

### XPointer - XML Pointer Language

[XPointer](#) enables addressing into the internal structures of [XML](#) documents. In particular, it provides for specific reference to elements, character strings, and other parts of [XML](#) documents, whether or not they bear an explicit [ID](#) attribute. [XPointers](#) can be used as fragment identifiers in [URI](#) references to specify a more precise sub-resource. Any [URI](#) fragment identifier that points into an [XML](#) resource must be an [XPointer](#). [XPointer](#) is often used together with [XLink](#), which specifies how to use hyperlinks with [XML](#).

### XSCS - XML Schema Compact Syntax

[XML Schema](#) is a rather complex [XML Schema Language](#), partly because of its inherent complexity, and partly because of its [XML](#) syntax. In an effort to reduce the syntactic verbosity and complexity of [XML Schema](#), [XSCS](#) defines a [EBNF](#)-based (and non-[XML](#)) syntax for [XML Schema](#). [XSCS](#) is designed for human users, and transformations from and to [XML Schema XML](#) syntax are implemented using [Java](#)-based tools.

## XSL - Extensible Style Language

[XSL](#) is a [Style Sheet Language](#) that can be used for displaying [XML](#) documents. Using [XSL](#) is two-step process, the first step being a transformation of the [XML](#) document using [XSLT](#), and the second step being the rendering of the result of the transformation, which is done using [XSL-FO](#). While [XSL](#) covers the same application area than [CSS](#), it is much more powerful, because the transformation step (using [XSLT](#)) can perform arbitrarily complex transformations of the [XML](#) document, while [CSS](#) is not able to make any structural changes to the [XML](#) document.

## XSL-FO - XSL Formatting Objects

[XSL-FO](#) is an [XML](#) vocabulary for the formatting of documents. Being part of [XSL](#), the normal way is to produce [XSL-FO](#) documents by transforming [XML](#) documents using [XSLT](#). Even though the principles behind [XSL](#) and [CSS](#) (the other [Style Sheet Language](#) created by [W3C](#)) are quite different, it is planned to align the formatting model between [XSL-FO](#) and [CSS](#), so that formatting engines can be based on the same code, both languages can be used to achieve the same results, and formatted results will look identical.

## XSLT - XSL Transformations

[XSLT](#) is a specialized [Programming Language](#) for transforming [XML](#) documents. Even though it is part of [XSL](#) and as such intended to be used for transforming [XML](#) documents into [XSL-FO](#) for presentation purposes, it is not limited to this application area. [XSLT](#) uses [XML](#) syntax (ie, it is a [Programming Language](#) in [XML](#) syntax), even though it is based on [DSSSL](#) (which uses a Lisp-like syntax). [XSLT](#) is particularly interesting in [B2B](#) scenarios, where [XML](#) documents must be transformed.

## XSLT Processor

An [XSLT Processor](#) is a base topic for topics describing programs which are [XSLT](#) processors. Basically, an [XSLT Processor](#) is a program interpreting and executing [XSLT](#) programs.

## XSP - Extensible Server Pages

[XSP](#) is [Cocoon](#)'s technology for building [WWW](#) applications based on dynamic [XML](#) content. An [XSP](#) page is a [Cocoon XML](#) document containing tag-based directives that specify how to generate dynamic content at request time. Upon [Cocoon](#) processing, these directives are replaced by generated content so that the resulting, augmented [XML](#) document can be subject to further processing (typically an [XSLT](#) transformation). [XSP](#) pages are transformed into [Cocoon](#) producers, typically as [Java](#) classes, though any scripting language for which a

Java-based processor exists could also be used. Directives can be either [XSP](#) built-in processing tags or user-defined library tags. [XSP](#) built-in tags are used to embed procedural logic, substitute expressions and dynamically build [XML](#) nodes. User-defined library tags act as templates that dictate how program code is generated from information encoded in each dynamic tag.

## Z39.50

[Z39.50](#) is a standard for information retrieval, it specifies a [Protocol](#) for the behaviour of two systems communicating for the purposes of database searching and information retrieval. As a network application standard, [Z39.50](#) is an open standard that enables communication between systems that run on different hardware and use different software. The [Z39.50](#) standard was developed to overcome the problems associated with multiple database searching. [Z39.50](#) simplifies the search process by making it possible for a searcher to use the familiar user interface of the local system to search both the local library catalogue as well as any remote database system that support the standard.