

Announcements

- ❖ No posting of code in the forum
- ❖ Check class announcements daily

Syndications Feed Formats

- ❖ **RSS 2.0**
 - ❖ Defines 30 elements
- ❖ **Atom 1.0**
 - ❖ Defines 21 elements
- ❖ Example of information using both formats
 - ❖ Search for “**Sample RSS and Atom Feeds**” in
 - ❖ <http://www.intertwingly.net/wiki/pie/Rss20AndAtom10Compared>
- ❖ Element Comparison Table
 - ❖ Search for “**Element Comparison Table**” in
 - ❖ <http://www.intertwingly.net/wiki/pie/Rss20AndAtom10Compared>

Semantic Web

- ❖ From:
 - ❖ <http://www.cs.umd.edu/~golbeck/LBSC690/SemanticWeb.html>
- ❖ Semantic web
 - ❖ Adds structure to the meaningful content of web pages
 - ❖ Information is given well-defined meaning
 - ❖ It will enable computers to be able to understand the data, instead of just displaying it
 - ❖ Most data in the web is for human consumption, rather than data that can be processed automatically by computers
- ❖ Allows for complex queries. For example:
 - ❖ How many stores are open when I am not at work or traveling in June?
- ❖ It is not a separate web, but an extension of our present web
- ❖ Semantic web – Term coined by Tim Berners-Lee

Semantic Web

- ❖ Semantic web needs
 - ❖ Access to structured collections of information
 - ❖ Sets of inference rules to conduct automated reasoning
 - ❖ Previous two referred to as knowledge representation
- ❖ Semantic Web Challenge
 - ❖ Provide a language that expresses both data and rules for reasoning about data
 - ❖ Allowing rules from knowledge-representation system to be exported to the web
- ❖ Technologies for developing the semantic web
 - ❖ XML
 - ❖ RDF (Resource Description Framework)
 - ❖ Microformats (<http://http://microformats.org/>)
 - ❖ hCard Creator → <http://microformats.org/code/hcard/creator>
 - ❖ hCalendar Creator → <http://microformats.org/code/hcalendar/creator>

Semantic Web: Ontology

- ❖ Document that formally defines relations among terms
- ❖ Typical ontology for the web has
 - ❖ Taxonomy
 - ❖ Inference rules
- ❖ Example of page marked up for ontology use:
 - ❖ <http://www.cs.rpi.edu/~hendler/>
 - ❖ Notice the <USE-ONTOLOGY> tag
 - ❖ Using ontology's concepts, information about where Dr. Hendler obtained his PhD is provided
 - ❖ Allow us to answer question: "Where Dr. Hendler received his degree?"

Semantic Web

- ❖ Impact of semantic web will be seen when:
 - ❖ Programs are created that combine information from different sources, process the information and exchange results with other programs
- ❖ Providing web-based services with semantics will
 - ❖ Allow for service discovery
 - ❖ Understanding of functions offered and how to take advantage of such functionality

Web Videos

- ❖ Introduction
 - ❖ <http://www.youtube.com/watch?v=OGg8A2zfWKg>
- ❖ Demo of Semantic Web Portal
 - ❖ <http://www.youtube.com/watch?v=X6WcpG5EEel&feature=related>
- ❖ Semantic Web of Data, Tim Berners-Lee
 - ❖ <http://www.youtube.com/watch?v=HeUrEh-nqtU&feature=related>
- ❖ Tim Berners-Lee: The next Web of open, linked data
 - ❖ http://www.youtube.com/watch?v=OM6XIIcm_go&feature=related

HTTP Protocol

- ❖ HTTP → Hypertext Transfer Protocol
 - ❖ Request-Response Networking Protocol based on a client-server computing model
 - ❖ A client submits a request to a server and the server provides resources. A response is sent by the server that includes status information and may include content requested by the client
 - ❖ Resources
 - ❖ Files (HTML, images)
 - ❖ Dynamically generated contents
 - ❖ Perform a specify task
- ❖ Clients → referred to as User Agents
 - ❖ Browsers
 - ❖ Spiders
- ❖ HTTP Resources are identified by Uniform Resource Identifiers
- ❖ Original version: HTTP/1.0, Revised: HTTP/1.1
- ❖ HTTP is a stateless protocol
- ❖ Standard way to of transferring information across the WWW (World Wide Web)

HTTP Protocol

- ❖ HTTP session
 - ❖ Sequence of network request-response transactions
- ❖ HTTP defines methods (**verbs**) that indicate the action to be performed on a specific resource
- ❖ Methods
 - ❖ **GET** → Requests a particular resource (e.g., a file)
 - ❖ **POST** → Submits data to be processed by a particular resource (e.g. inputting data into a database)
 - ❖ **HEAD** → Returns what GET returns, but without the actual resource (just headers)
 - ❖ **PUT** → Sends a representation of a particular resource
 - ❖ **DELETE** → Deletes the specified resource
 - ❖ **TRACE** → Echoes received request so client can check if any changes have been made
 - ❖ **OPTIONS** → HTTP method supported by the server for the specified URL
 - ❖ **CONNECT** → Converts request connection to TCP/IP tunnel
 - ❖ **PATCH** → To apply partial modifications to a resource
- ❖ HTTP servers must implement GET and HEAD and whenever possible OPTIONS
- ❖ Response status codes

HTTP Protocol

- ❖ Response status codes (five classes)
 - ❖ 1xx Informational
 - ❖ 2xx Success
 - ❖ 200 → OK
 - ❖ 3xx Redirection
 - ❖ 4xx Client Error
 - ❖ 401 → Unauthorized
 - ❖ 404 → Not found
 - ❖ 5xx Server Error
- ❖ Full listing at:
 - ❖ [http://en.wikipedia.org/wiki/List of HTTP status codes](http://en.wikipedia.org/wiki/List_of_HTTP_status_codes)
- ❖ Example (using telnet)
 - ❖ <http://www.cs.umd.edu/class/fall2005/cmsc433/HowWebServersWork.html>