

Definitions and Concepts

hussein suleman
uct cs honours 2006

Non-IT Examples

- Why do the staples always fit in the stapler? Do they always fit?
- Why does your TV always work with your VCR? Does it always work?
- Have you ever bought shoes? Does the size always work? Why or why not?
- How come a paper punch always makes holes at the same distance apart? Does it always?



IT Example

- Pay for groceries at Grab `n Go
 - Teller swipes credit card
 - Machine knows standard card format
 - Machine contacts store's bank
 - Store's bank contacts customer's bank
 - Customer's bank checks account, deducts sum and transfers to store's bank
 - Store's bank is given clearance to proceed
 - Machine prints out receipt



Types of Interoperability

- Software System Interoperability
- Business Process Interoperability
- Manufacturing
- Networking (Hardware) Interoperability

Interoperability

Software

With respect to **software**, the term interoperability is used to describe the capability of different programs to exchange data via a common set of business procedures, and to read and write the same **file formats** and use the same **protocols**.

(The ability to execute the same **binary code** on

source: <http://en.wikipedia.org/wiki/Interoperability>

Why Interoperate?

- Connect services together
 - e.g., Transfer money from one bank to another electronically
- Build meta-archives
 - e.g., Google Scholar
- Save money on development
 - e.g., Well-known DOM APIs and tools
- Preserve information and increase accessibility
 - e.g., PDF(/A) as a standard data format
- Promote openness/choice
 - e.g., Microsoft's support for Open Document Format!



Syntactic vs. Semantic Interoperability

- Syntactic means the **syntax** is standardised.
 - e.g., XML
- Semantic means the **meaning** is standardised.
 - e.g., Dublin Core metadata
- Many popular standards define abstract semantics and one or more concrete syntaxes.
 - e.g., Dublin Core

Data Formats vs. Protocols

- Data Formats refer to how data is stored.
 - Can we standardise data formats?
 - How much can we specify in terms of syntax and semantics?
- Protocols refer to how communication occurs between parties.
 - Can we standardise communications between 2 parties?
 - Can we design standard protocols for large groups of parties with common interests?
- How do we build solutions with both protocols and standard data formats?
 - Where does the protocol end and the data format start?

Standards vs. Specifications

- A Specification can be any formal description.
 - Any non-trivial software tool usually has specifications.
- A Standard is a formal description that is endorsed, validated and maintained by a standards body.
 - e.g., HTTP, endorsed by the IETF/W3C

```
Network Working Group  
Request for Comments: 2616  
Obsoletes: 2068  
Category: Standards Track
```

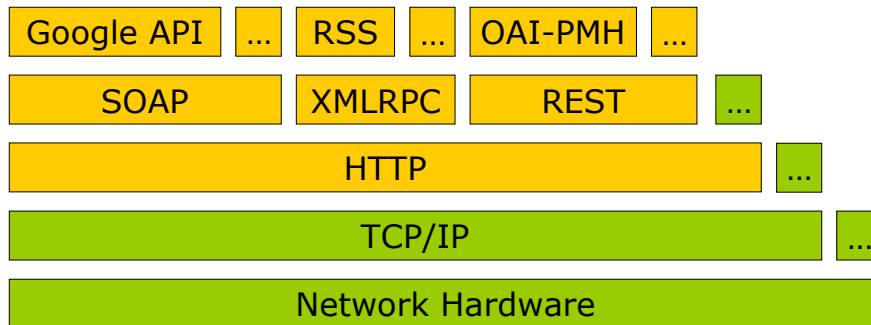
source: <ftp://ftp.rfc-editor.org/in-notes/rfc2616.txt>

- You can rely on standards, but not on specifications!

Degrees of Interoperability

- How much interoperability do you need?
 - Syntactic?
 - e.g., Remote access to data.
 - Semantic?
 - e.g., Standard metadata formats.
 - Batch or Real-Time?
 - How much semantics to encode in data?
 - How many services/interfaces?
 - What transports?
 - HTTP/SOAP/OAI-PMH/...?

Layered Web Interoperability



Cathedral vs. Bazaar

- Cathedral: One person sets out a specification and everyone must use it.
 - e.g., Google API
- Bazaar: Everyone agrees to a common standard, then everyone implements it.
 - e.g., HTTP

Trust and Security

- Interoperability is only possible among trusted parties.
- Issues:
 - Authentication
 - Authorisation
 - Encryption
 - Privacy
 - Trust

Validation

- To what degree does an implementation follow a standard?
- Data validation:
 - Does an XML-encoded data stream conform to its DTD? XML Schema?
 - Is the data of a high quality?
 - Is the encoding correct?
- Protocol validation:
 - Is the protocol implementation correct? Robust? Reliable? Consistent?

Protocol Testing

The [OAI-PMH](#) defines a set of requests and these requests, specified through an [XML schema](#) respond to each of the protocol requests with a schema. It also must respond to mal-formed requests with [exception conditions](#).

Conformance Testing for Basic F

A repository must successfully complete the registry:

For every protocol request, the repository re

- ◆ is valid XML (the XML successfully p

source: http://www.openarchives.org/data/registerasprovider.html#Protocol_Conformance_Testing

Open *

to all, without restriction ←

- Open Source,
- Open Standards
- Open Content
- Open Archives
- Open Access

1.2 Document Scope

The software systems discussed here satisfy three criteria:

They are available under an Open Source license—that is, they are available for free and can be freely modified, upgraded, and redistributed.

They comply with the latest version of the Open Archives Initiative metadata harvesting protocols—this OAI compliance helps ensure that each implementation can participate in a global network of interoperable research repositories. And,

They are currently released and publicly available—several new systems are currently being developed. As these systems become available for public release, we will revise this guide to include them.

source: http://www.soros.org/openaccess/pdf/OSI_Guide_to_IR_Software_v3.pdf

References

- Crow, R. (2004) A Guide to Institutional Repository Software, OSI. Available http://www.soros.org/openaccess/pdf/OSI_Guide_to_IR_Software_v3.pdf
- Fielding, R., J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach and T. Berners-Lee (1999) Hypertext Transfer Protocol – HTTP/1.1, RFC 2616, Network Working Group, IETF. Available <ftp://ftp.rfc-editor.org/in-notes/rfc2616.txt>
- Open Archives Initiative (2005) Data Provider Registration. Website http://www.openarchives.org/data/registerasprovider.html#Protocol_Conformance_Testing
- Wikipedia (2006) Interoperability. Website <http://en.wikipedia.org/wiki/Interoperability>