

**CS 683 Emerging Technologies
Spring Semester, 2003
Doc 19 REST
Contents**

SOAP – Some Numbers 2
REST 3

References

<http://www-106.ibm.com/developerworks/webservices/library/ws-pyth9/#table1>

REST Tutorial <http://www.xfront.com/sld001.htm>

Fielding & Taylor paper: Principled Design of the Modern Web Architecture, ACM Transactions on Internet Technology, Vol. 2, No. 2, May 2002,
<http://www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf>

2003 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA.
OpenContent (<http://www.opencontent.org/opl.shtml>) license defines the copyright on this document.

SOAP – Some Numbers

<http://www-106.ibm.com/developerworks/webservices/library/ws-pyth9/#table1>

Mike Olson (mike.olson@fourthought.com), Principal Consultant, Fourthought, Inc.
Uche Ogbuji (uche.ogbuji@fourthought.com), Principal Consultant, Fourthought, Inc.

Technology	Connect time	Send string (21,000 characters)	Receive string (22,000 characters)	Send 5,000 integers
Raw sockets	0.002242	0.001377	0.001359	6.740674
CORBA	0.000734	0.004601	0.002188	1.523799
XML-RPC	0.007040	0.082755	0.050199	100.337219
SOAP	0.000610	0.294198	0.279341	1,324.296742

REST

<http://developers.slashdot.org/article.pl?sid=03/04/03/1942235&mode=nocomment&tid=185&tid=156>

tadghin:

“I was recently talking with Jeff Barr, creator of [syndic8](#) and now Amazon's chief web services evangelist. He let drop an interesting tidbit. Amazon has both [SOAP](#) and REST interfaces to their web services, and 85% of their usage is of the REST interface.”

“ Despite all of the corporate hype over the SOAP stack, this is pretty compelling evidence that developers like the simpler REST approach. “

REST Resources

REST Tutorial <http://www.xfront.com/sld001.htm>

REST Wiki <http://internet.conveyor.com/RESTwiki/moin.cgi/>

Fielding Ph D. Thesis: Architectural Styles and the Design of Network-based Software Architectures, 2000,
<http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

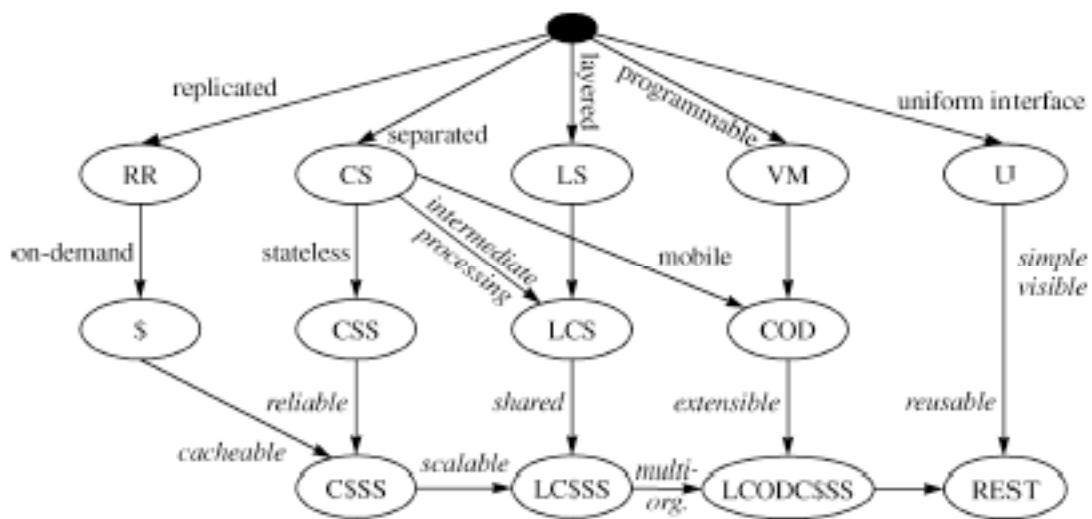
Fielding & Taylor paper: Principled Design of the Modern Web Architecture, ACM Transactions on Internet Technology, Vol. 2, No. 2, May 2002,
<http://www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf>

REST – Representation State Transfer

Web Requirements

- Low entry barrier
- Extensible
- Distributed Hypermedia
- Internet-scale
 - Anarchic Scalability
 - Independent deployment
- Evolving Requirements

REST as Hybrid Style



- Client-Server
- Stateless
- Cache
- Layered Systems
- Uniform Interfaces
- Code on Demand

REST Architecture Elements

Data Elements

Distributed Objects style – hide data in processing components

Distributed Hypermedia

- Server renders data send client fixed-format message
- Send data and rendering engine to client
- Send data and metadata – let client select rendering engine

REST uses hybrid of all three

Resources & Resource Identifiers

Representation

Connectors

Client

Server

Cache

Resolver

Tunnel

Example

www.amazon.com

Tutorial

<http://www.xfront.com/sld001.htm>