# Introduction

This is a test of Office 2008 for the Mac and generating *XML content*. The idea is to show students that XML is all around them by pointing out to them what the x at the end of the extension means and how they can find out things from the document structure. Then we move on to Web services via REST and then SOAP so that they can see XML in practice that is useful.

## Scraping HTML pages

## HTML Forms

Beyond scraping information out of HTML files, we can send requests to servers via mimicking the actions of HTML forms. The result is still an HTML document (typically).

## REST

REST is Representational State Transfer and allows us to send a request to a URL with parameters like an HTML form and whose result is the representation of the resource (at that time, at least). The result can use any format but either XML or JSON is typical. This gives us much more structured information.

Rest is becoming quite popular because it is quite straightforward to understand and get started. All one needs is the ability to submit a request to a URL via either HTPP GET or POST actions. The RCurl package allows us to do this in R via getForm() and postForm().

### Example Sites

Example REST service sites include

* Yahoo Geo Location
* [Wunderground weather](http://www.wunderground.com)
* NOAA weather forecast
* [MusicBrainz](http://www.musicbrainz.com)

## SOAP

SOAP stands for Simple Object Access Protocol. It is a richer and more complicated approach to Web services. It allows us to send richer arguments/objects as parameters to Web service methods. Instead of simple strings, we can send complex objects such as structures with fields which are simple or complex objects, matrices, data frames, … SOAP typically uses HTTP to transmit the request. It uses XML to format the request and arguments, and to return the result.

SOAP is also strongly associated with WS\* technologies. This includes WSDL (Web Service Descrption Language) which allows servers to describe all of the methods and the data types used within them via an XML file. This allows clients to read the information and programmatically generate client-side interfaces to invoke those methods. Users don’t have to know anything about SOAP to be able to access the service’s methods.