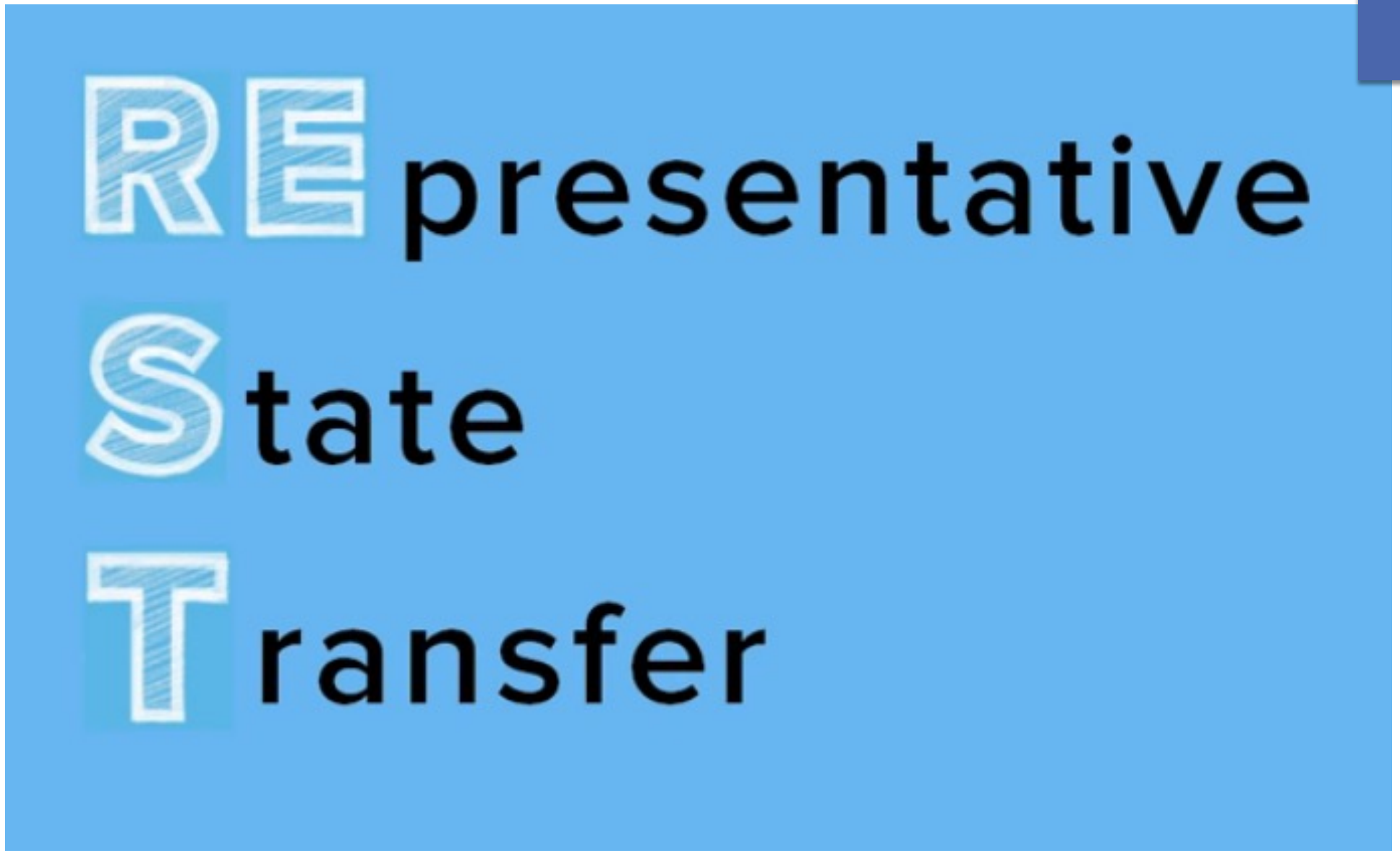
WEB SERVICES - 2

WEEK 11 – 03/24/2021


REpresentative
State
Transfer

University of Colorado Boulder

REST IS AN *ARCHITECTURAL STYLE*

Modern Architectural Style:

Colonial Architectural Style:

WEB SERVICES

The "architectural style" is an abstract concept - it defines the characteristics and features you would find in a house built according to that style

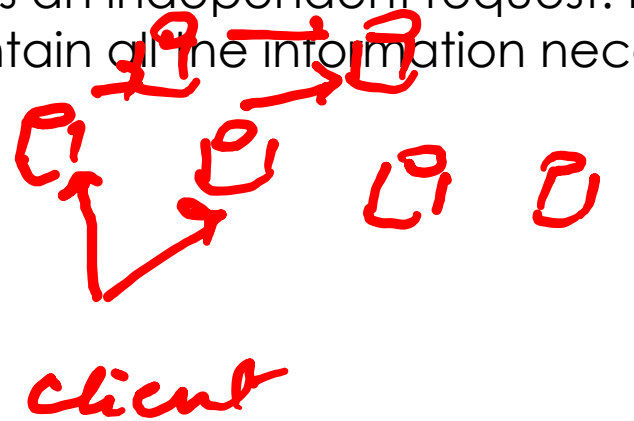
It is NOT the same as the house itself.

REST is an abstract concept that defines the characteristics and features you would find in a web service request built according to the REST style

REST is not really a protocol – it is a set of standards used to define Web Services

WEB SERVICES

- Everything in REST is considered as a resource.
  - Every resource is identified by an URI.
  - Uses uniform interfaces. Resources are handled using http POST, GET, PUT, DELETE operations
  - Stateless. Every request is an independent request. Each request from client to server must contain all the information necessary to understand the request.



University of Colorado Boulder

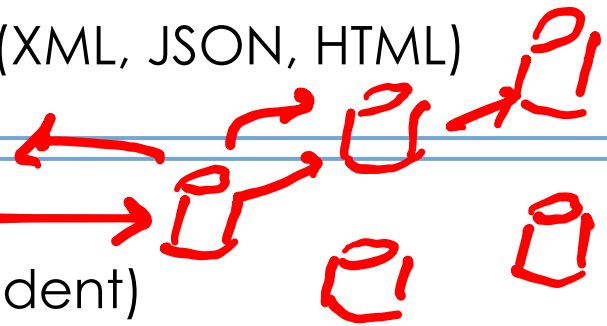
WEB SERVICES

A RESTFul web service typically defines the base URI for the services, the format/rules of the API, and the set of operations (POST, GET, PUT, DELETE) are supported, when HTTP is used.

University of Colorado Boulder

CHARACTERISTICS OF A REQUEST/RESPONSE FOLLOWING THE REST STYLE

Resources follow the rules	URI (identifies the resource being requested) Uniform Interface Methods (GET, PUT, POST, etc.) Uniform Interface Representation (XML, JSON, HTML)
Protocols offer features	Client-Server (like HTTP) Stateless (each request is independent) Layered (may pass through intermediaries) Cacheable (intermediaries may cache for performance)



University of Colorado Boulder

ADVANTAGES OF A REQUEST/RESPONSE FOLLOWING THE REST PROTOCOL

Efficiency

(through caching & compression)

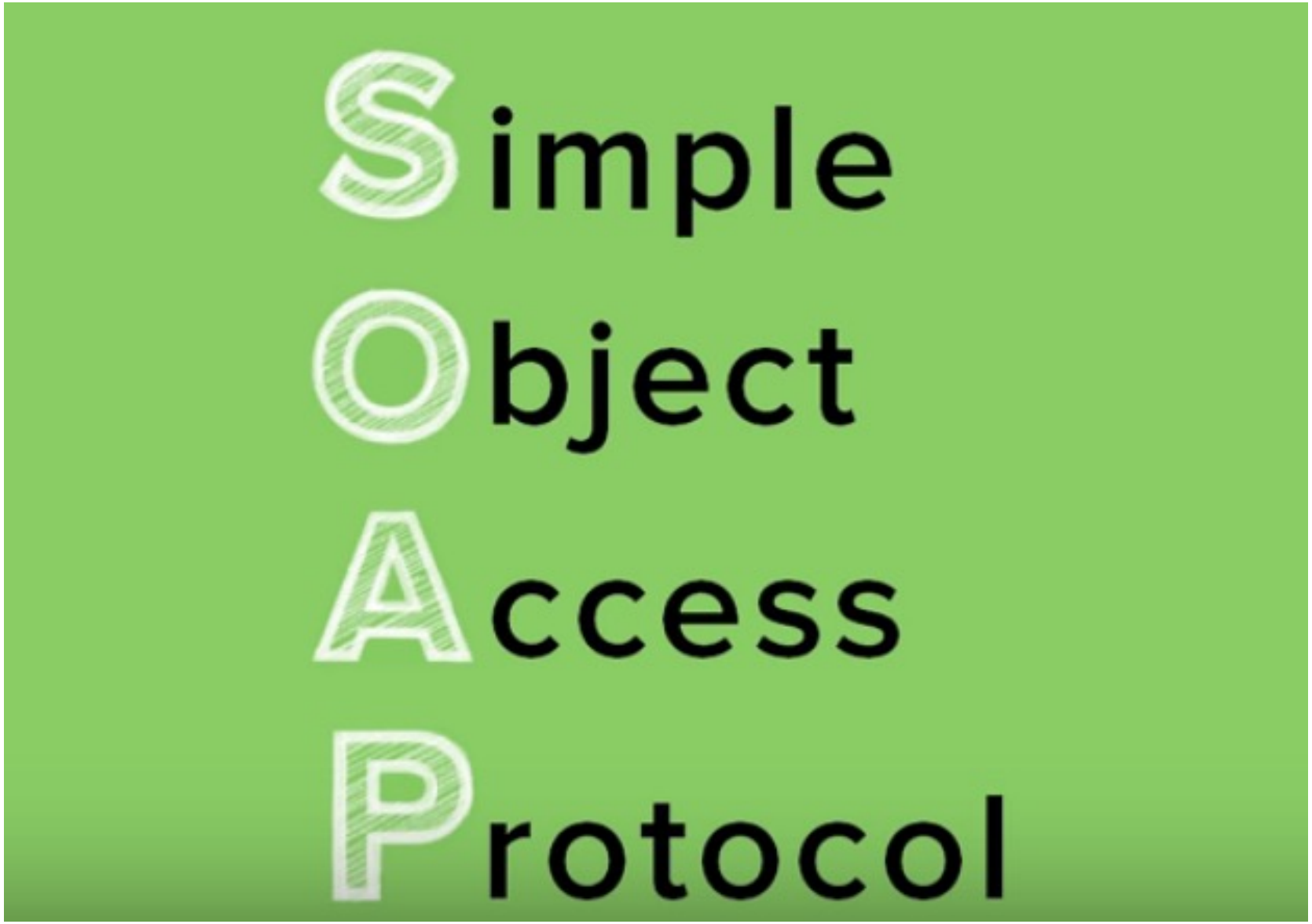
Scalability

(gateways distribute traffic, caching, statelessness allows different intermediaries)

User Perceived Performance

(code on demand, client validation, caching)

Simplicity


Simple
Object
Access
Protocol

University of Colorado Boulder

WEB SERVICES

REST

REST (Representational State Transfer) was Created in 2000 by Roy Fielding in UC, Irvine. Developed in an academic environment, this protocol embraces the philosophy of the open Web.

SOAP

SOAP (Simple Object Access Protocol), was created in 1998 by Dave Winer et al in collaboration with Microsoft. Developed by a large software company, this protocol addresses the goal of addressing the needs of the enterprise market.

SOAP

- A SOAP message is an XML document containing the following elements:
  - An **Envelope** element that identifies the XML document as a SOAP message
  - A **Header** element that contains header information
  - A **Body** element that contains call and response information
  - A **Fault** element containing errors and status information

https://en.wikipedia.org/wiki/SOAP

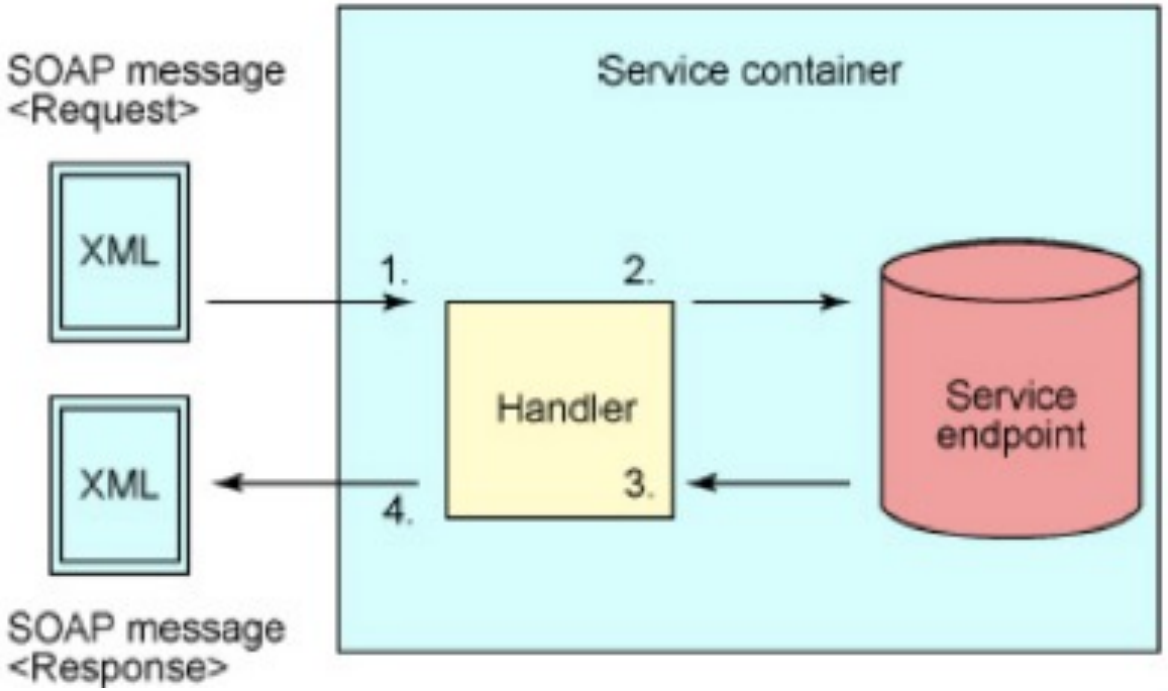
SOAP

WSDL (Web Service Description Language) is an XML document that defines "contract" between client and service and is static by its nature.

SOAP builds an XML based protocol on top of HTTP or some other protocol according to the rules described in the WSDL for that Web Service.

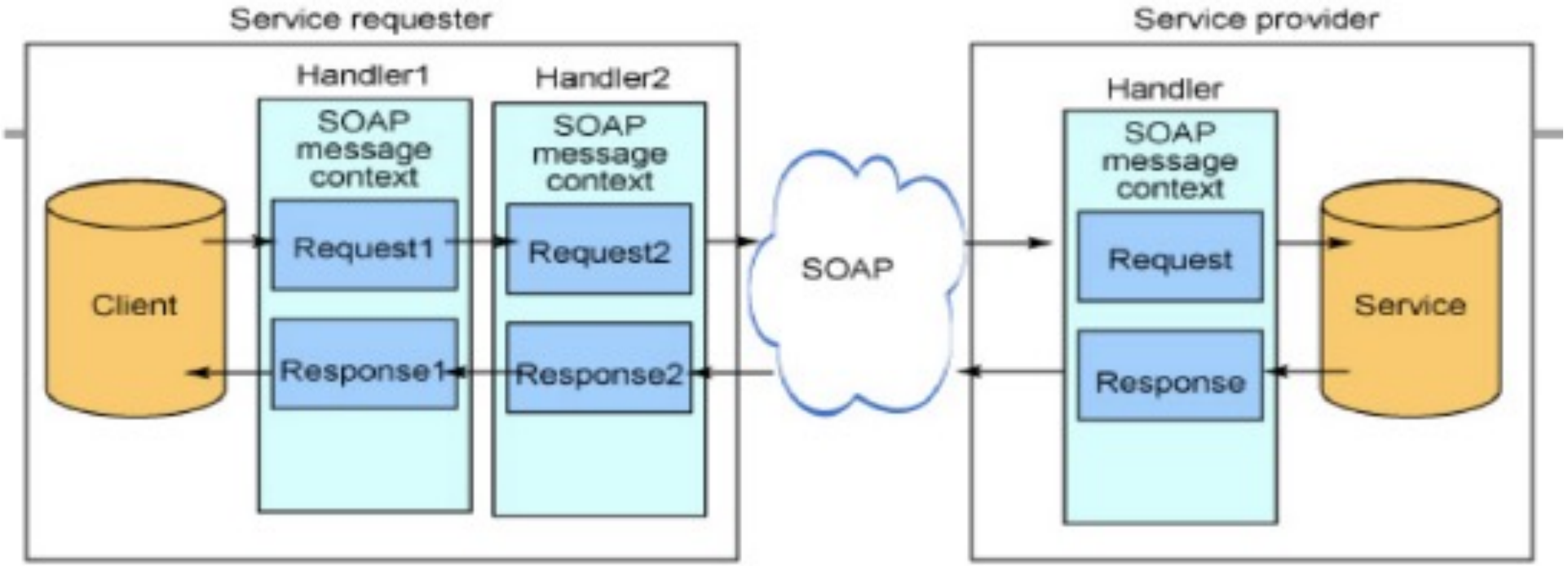
SOAP Handlers

- **Handlers** are pluggable classes that can be associated with a Web service or Web service client to provide pre-processing or post-processing of XML messages.
  - Ex: logging XML traffic through a Web service
  - Ex: measure performance by accessing the SOAP header to insert initial and finish times between two checkpoints



University of Colorado Boulder

SOAP



University of Colorado Boulder

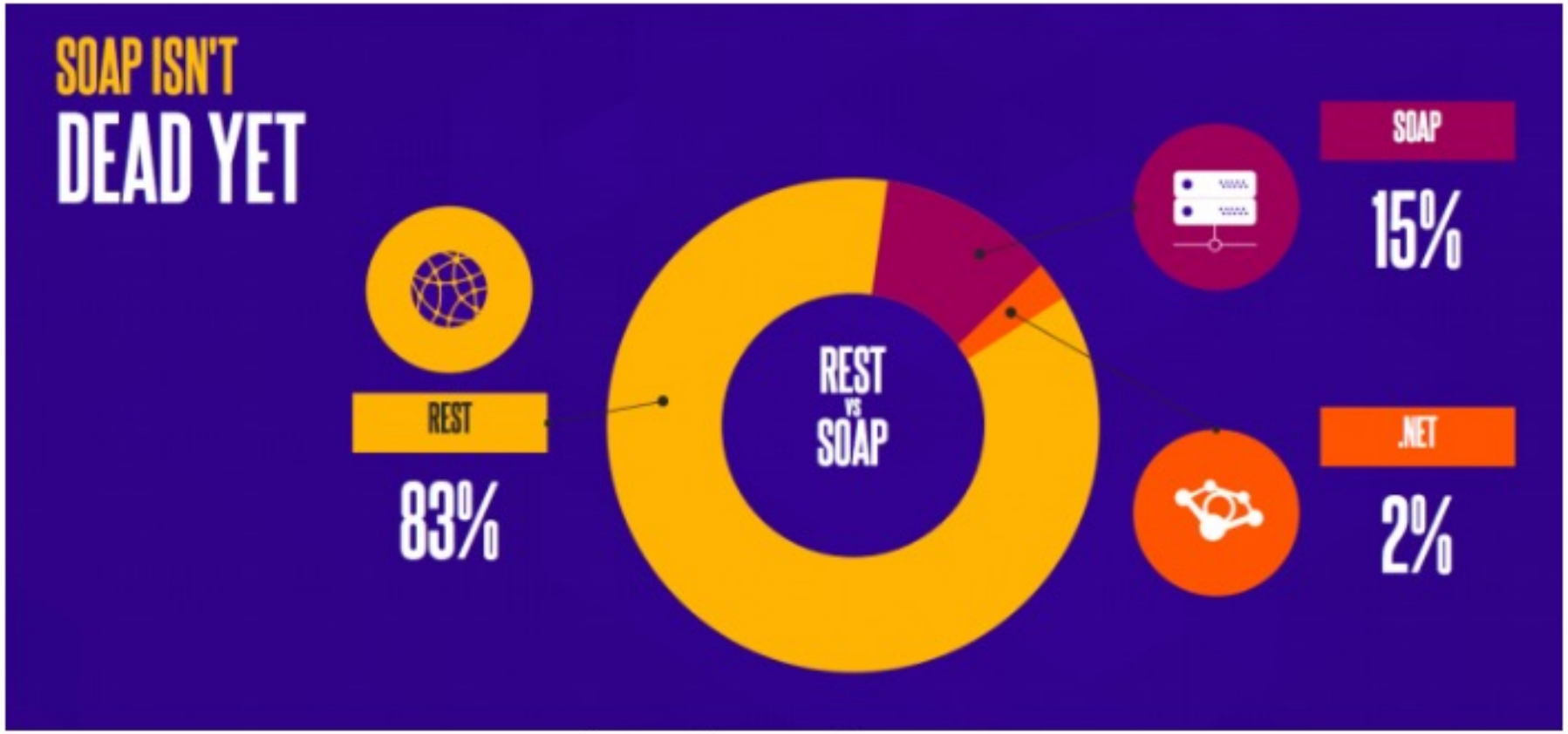
WEB SERVICES

REST	SOAP
Representational State Transfer	Simple Object Access Protocol
Architecture Style	An actual protocol
Uses simple HTTP	Uses SOAP envelope, then HTTP (or FTP, or other) to transfer the data
Uses many different data formats like JSON, XML, YAML*	Supports only XML format
Performance & Scalability & Caching	Slower performance. Scalability is limited and complex. Caching is not possible.
Widely and frequently used	Used where REST is not possible

*YAML: YAML Ain't Markup Language

What Is It: YAML is a human friendly data serialization standard for all programming languages.

According to this year's report, one area that continues to come up year after year is the balance of power between SOAP and REST. Although REST dominates the scene, there is still a decent percentage SOAP APIs out there that can't be ignored – at least 15% based on Cloud Elements' experience.



The State of API Integration report 2017

University of Colorado Boulder

DEMO

- We'll see some examples of APIs and the data we can fetch and display from them.