INTEROPERABILITY

IS THE ABILITY OF TWO OR MORE SYSTEMS OR COMPONENTS TO EXCHANGE INFORMATION AND TO USE THE INFORMATION THAT HAS BEEN EXCHANGED.

NOWDAYS IT'S VERY IMPORTANT BECAUSE THE VARIOUS SYSTEMS MUST INTERACT AND SHARE DATA WITH EACH OTHER. INTEROPERABILITY FACILITATES TRANSACTIONS AND DATA MANAGEMENT ACROSS DIFFERENT PLATFORM. IT CAN ALSO IMPROVE EFFICIENCY AND REDUCE COST IN VARIOUS INDUSTRIES.

SOME PLATFORM OFFER THE POSSIBILITY OF USING THEIR FUNCTIONS THROUGH APIs.

LEVEL AND TASKS

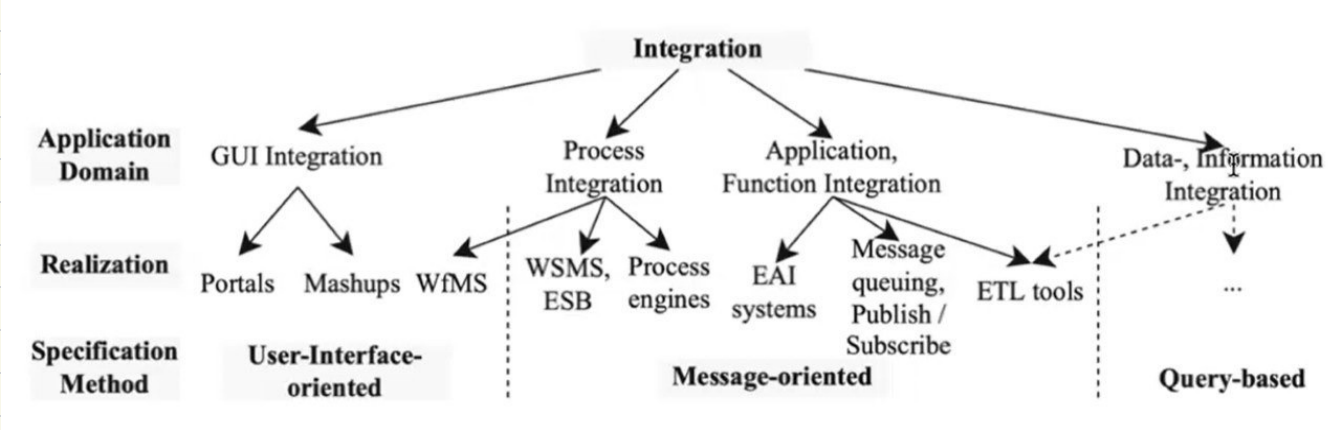
Levels ↝/tasks ↡	Syntactical	Semantic	Organizational
Exchange	Exchange formats, e.g., XML, relational databases, JSON, BSON, YAML, MessagePack; query languages: XPath, XQuery, SQL; transformation languages, e.g., XSLT	Schema matching and mapping; ontologies	Message exchange; correlation
Integrate	SQL/XML standard; native XML databases; REST and GraphQL	Edge table, shredding; XML schema and RNG; schema and data integration; service integration	Correlation and choreography
Orchestrate	BPMN, Petri Nets, Workflow Nets, RPST, CPEE Trees, Colored Petri Nets	Verification; task and worklist design; service invocation; correlation; integration patterns and processes	Choreography

EXCHANGE: THE NEED TO BE ABLE TO EXCHANGE INFORMATION AND KNOW THE SENDER OF THE INFORMATION.

INTEGRATE: THE NEED FOR INTEGRATION OF DATA FROM DIFFERENT SELLERS AND THE NEED FOR A COMMON REPRESENTATION OF INFORMATION. THE NEED FOR A SYSTEM THAT CAN HANDLE DIFFERENT TYPES AND FORMATS OF DATA, AS WELL AS THE IMPORTANCE OF INTEGRATING DATA FROM DIFFERENT SOURCES.

ORCHESTRATE: THE NEED FOR ORCHESTRATION AND COORDINATION DUE TO THE COMPLEXITY OF INTEGRATING DIFFERENT ORGANIZATIONAL SYSTEMS. THIS LEVEL IS ORGANIZED THROUGH CHOREOGRAPHY. CHOREOGRAPHY IS HOW TO COORDINATE DIFFERENT ORGANIZATIONS SUCH THAT THEY EXCHANGE DATA IN ORDER TO HAVE THE SAME VIEW.

INTEGRATION SCENARIOS AND METHODS



INTEGRATION IS DIVIDED INTO FOUR MAIN APPLICATION DOMAINS:

- **GUI INTEGRATION**: THE AIM IS TO UNIFY THE GRAPHICAL INTERFACES OF DIFFERENT SYSTEMS TO OFFER USERS A SINGLE EXPERIENCE BY AGGREGATING CONTENT OR FUNCTIONALITY FROM MULTIPLE SYSTEMS.

  USER-INTERFACE-ORIENTED

- **PROCESS INTEGRATION**: THE PURPOSE IS TO CONNECT AND SYNCHRONIZE BUSINESS OR OPERATIONAL PROCESSES BETWEEN MULTIPLE SYSTEMS.

- **APPLICATION, FUNCTION INTEGRATION**: THE PURPOSE IS TO ALLOW DIFFERENT APPLICATIONS OR MODULES OF A SYSTEM TO COMUNICATE AND WORK TOGETHER.

  MESSAGE-ORIENTED

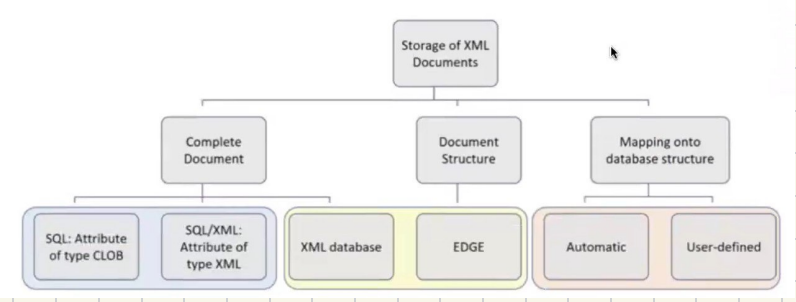
- **DATA, INFORMATION INTEGRATION**: THE PURPOSE IS TO UNIFY AND MANAGE DATA FROM DIFFERENT SOURCES TO ENSURE CONSISTENCY AND ACCESSIBILITY.

  QUERY-BASED

STANDARDS

- **XML**: USEFUL TO EXCHANGING INFORMATION
- **XLINK**: ALLOWS TO REFERENCE ELEMENT, THE SAME OR OTHER XML DOCUMENTS.
- **XPOINTER**: ALLOWS TO POINT TO PARTS OF THE CONTENT OF AN XML DOCUMENT BY UTILIZING THE XPATH STANDARD.
- **XINCLUDE**: COMBINE MULTIPLE DOCUMENTS/FRAGMENTS INTO ONE.

XLINK AND XPOINTER ARE USUALLY COMBINED FOR CREATING REFERENCES TO PARTS OF OTHER DOCUMENTS.



- **JSON**: IS SIMPLE TREE-BASED SERIALIZATION FORMAT THAT FOCUSES ON A FEW SIMPLE DATA TYPES PLUS ARRAY AND DICTIONARIES.
- **BSON**. A MORE COMPACT VERSION OF JSON.
- **YAML**: FOCUSES ON THE SAME CONCEPTS AS JSON BUT ACHIEVES NESTING BY INDENTATION.
- **MESSAGEPACK**: OPTIMIZED TOWARD COMPACTNESS AND DE/SERIALIZATION SPEED.

Format	Length	SP	DP
XML	310	31.92	11.24
JSON-PP	194	1.59	1.48
JSON	130	1.59	1.48
BSON	129	1.60	1.81
YAML	140	73.58	13.82
MSGPACK	97	1.00	1.00

SERVICE INTEROP

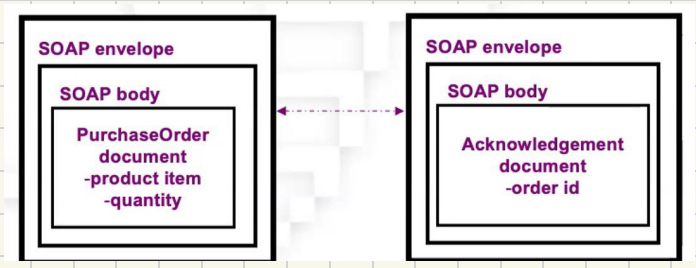
SERVICES SOLVES TWO INTEROP ASPECTS: A SERVICE PROVIDES ABSTRACT ACCESS TO COMPLEX FUNCTIONALITY OR DATA, AND FOSTERS REUSABILITY.

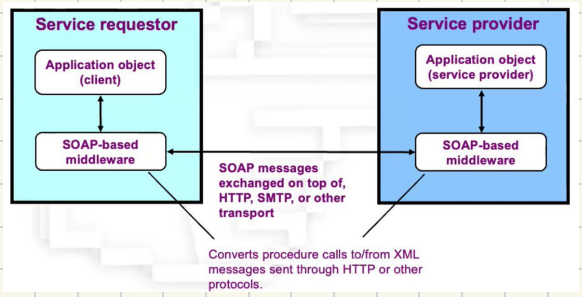
SERVICE ORIENTED ARCHITECTURES (SOAs) ARE ONE OF THE STAPLES OF INTEROPERABLE SW DESIGN, AS THEY FACILITATE THESE IMPORTANT PROPERTIES:

- **LOOSE COUPLING**: A CLEAN AND MINIMAL API DESIGN IS RESPONSIBLE FOR THE DEGREE OF COUPLING. THIS LEADS TO LESS COMPLICATED DEPENDENCIES IN THE CODE, LESS CRASHES AND BETTER MAINTAINABILITY.
- **SCALABILITY**: COMPONENTS CAN BE DISTRIBUTED TO DIFFERENT NODES, CAN EMPLOY LOAD-BALANCING AND FAIL-OVER.
- **REUSABILITY**: INDIPENDENT COMPONENTS ARE PROVIDED FOR OTHER SW DEVELOPMENT TEAMS TO BE USED WITHOUT ANY EXTRA INSTALLATION COSTS.
- **FLEXIBILITY, MODULARITY, AGILITY**: SOAs PROMOTE THESE PROPERTIES BY PROVIDING INDEPENDENT COMPONENTS, WHICH CAN BE CHANGED AND ADAPTED TO CHANGING REQUIREMENTS INDEPENDENTLY.

SOAP

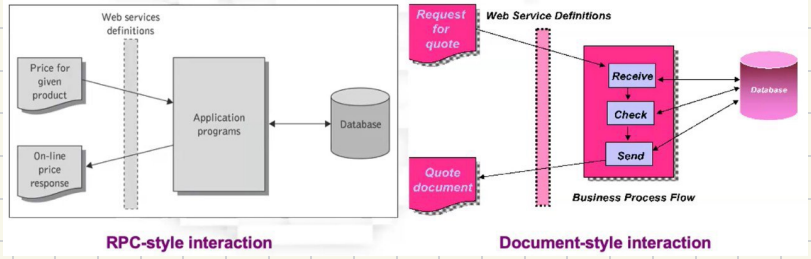
IS THE STANDARD MESSAGING PROTOCOL USED BY WEB SERVICES. IT CODIFIES THE USE OF XML AS AN ENCODING SCHEME FOR A REQUEST AND RESPONSE PARAMETERS USING HTTP AS A MEANS FOR TRANSPORT.





SUPPORTS TWO POSSIBLE COMMUNICATION STYLES:

- REMOTE PROCEDURE CALL (RPC).
- DOCUMENT (OR MESSAGE).



Advantages of SOAP are:

- Simplicity
- Portability
- Firewall friendliness
- Use of open standards
- Interoperability
- Universal acceptance.

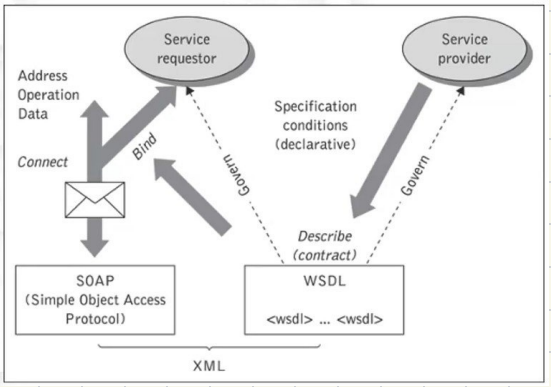
Disadvantages of SOAP are:

- Too much reliance on HTTP
- Statelessness
- Serialization by value and not by reference.

WEB SERVICE

SERVICE DESCRIPTION REDUCES THE AMOUNT OF REQUIRED COMMON UNDERSTANDING AND CUSTOM PROGRAMMING AND INTEGRATION.

THE WEB SERVICES DESCRIPTION LANGUAGE (WSDL) IS THE XML BASED SERVICE REPRESENTATION LANGUAGE USED TO DESCRIBE THE DETAILS OF THE COMPLETE INTERFACES EXPOSED BY WEB SERVICES AND THUS IS THE MEANS TO ACCESSING A WEB SERVICE.



RESTFUL SERVICES

REST REFERS TO SIMPLE APPLICATION INTERFACES TRASMITTING DATA OVER HTTP WITHOUT ADDITIONAL LAYERS AS SOAP.

REST IS A KIND OF RPC, EXCEPT THE METHODS HAVE BEEN DEFINED IN ADVANCE.