

“Get into SWIM” session

Service Design and Implementation

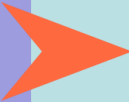
Pedro Fernandez

DECMA / RTD / DAI - Digitalisation and Information unit

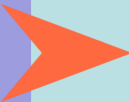
22nd May 2019

Design

service design is the set of activities involved in expressing what the service does and how it works. ... typically use a modelling language notation (e.g. UML) to represent the blueprint of the information service as a service model (e.g. including service interfaces, service operations, and service payload).



Selection of the message exchange pattern (MEP)



Definition of the service (interface, service operations and information service payload)

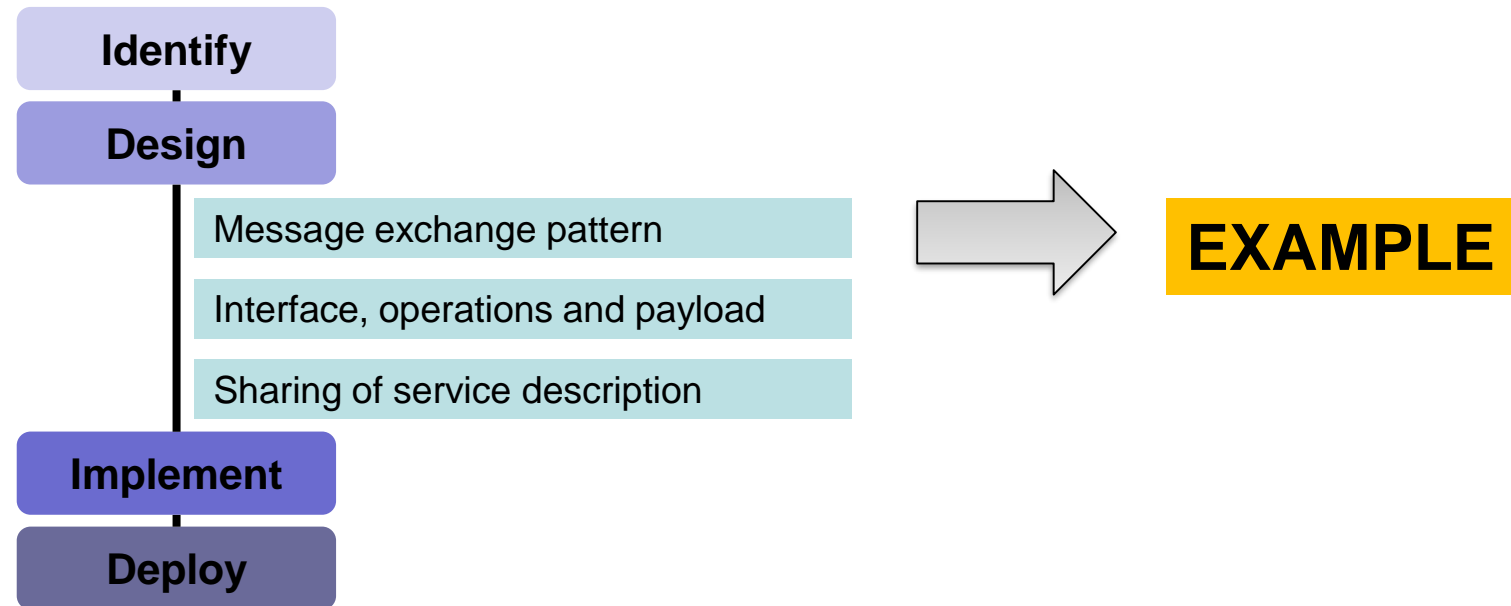
- Sharing of service description information (e.g. using a SWIM Service Registry)



- Chosen MEP.
- Service model (service interface(s), service operation(s), service behaviour)
- Service payload (the logical). representation of the information exchanged by the service interface operations
- Service design information expressing what the service does and how it works



MEP Identification



References

Resources



SWIM TI Message Exchange Patterns Identification Guidelines

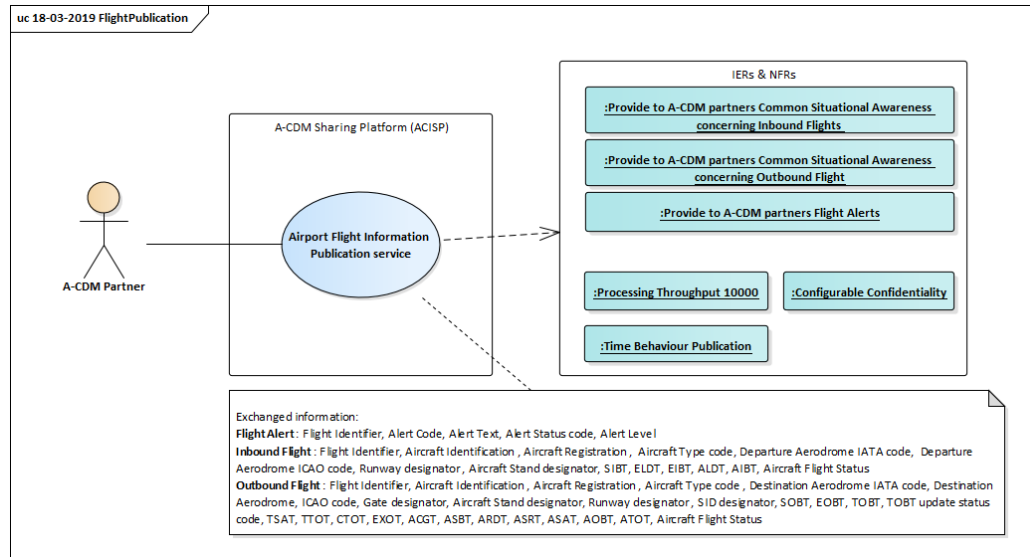
<https://ost.eurocontrol.int/sites/AISWIM/SWIMspecs/TEC/Shared%20Documents/Supporting%20Material/02%20-%20SWIM%20TI%20MEPs%20Identification%20Guidelines/SWIM%20TI%20Message%20Exchange%20Patterns%20Identification%20Guidelines.docx>

Overview

Cardinality	1-1/1-N/N-M
Space Decoupling	YES/NO
Time Decoupling	YES/NO
Process Decoupling	Synchronous/Asynchronous

- One Way
- Synchronous Request/Reply
- Asynchronous Request/Reply
- Fan-out
- Publish/Subscribe Push
- Publish/Subscribe Pull
- Brokered Publish/Subscribe Push

MEP Related Characteristics Review



Cardinality	1-N
Space Decoupling	NO/YES
Time Decoupling	NO/YES
Process Decoupling	Asynchronous



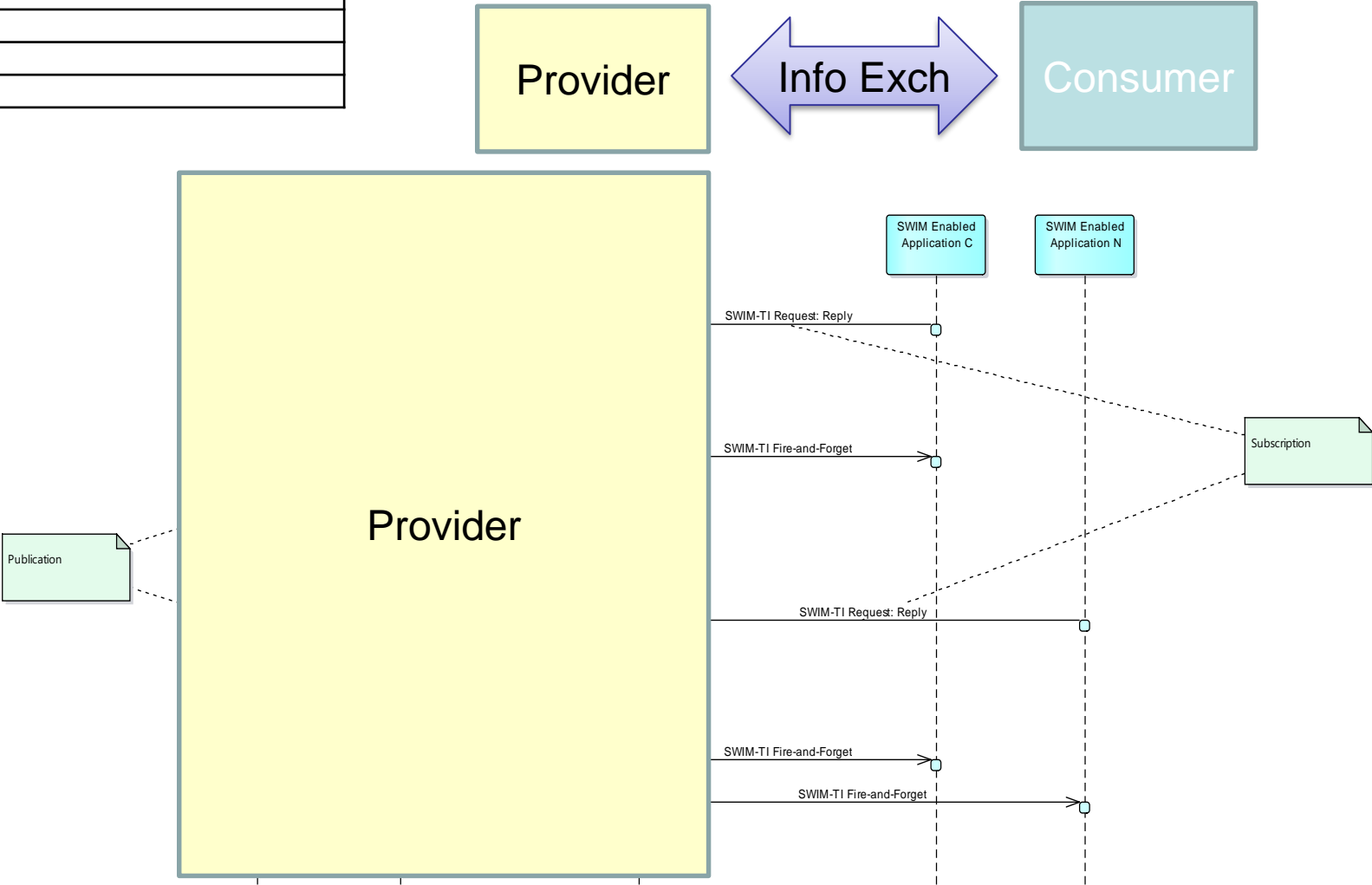
MEP Catalogue Review

Cardinality	1-N
Space Decoupling	NO/YES
Time Decoupling	NO/YES
Process Decoupling	Asynchronous

- ~~One Way~~
- ~~Synchronous Request/Reply~~
- ~~Asynchronous Request/Reply~~
- ~~Fan-out~~
- **Publish/Subscribe Push**
- Publish/Subscribe Pull
- **Brokered Publish/Subscribe Push**

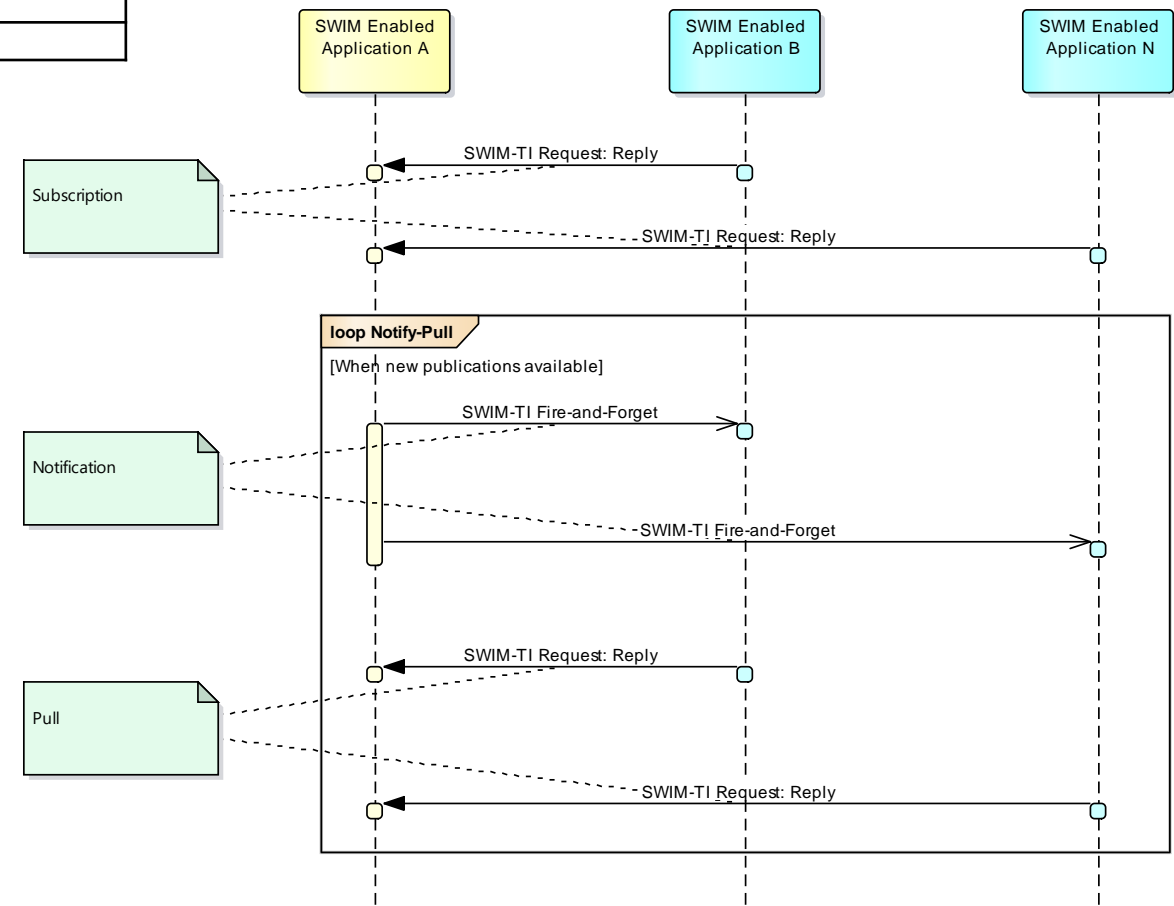
Brokered Publish/Subscribe Push

Cardinality	Many-many
Space Decoupling	YES
Time Decoupling	YES
Process Decoupling	Asynchronous



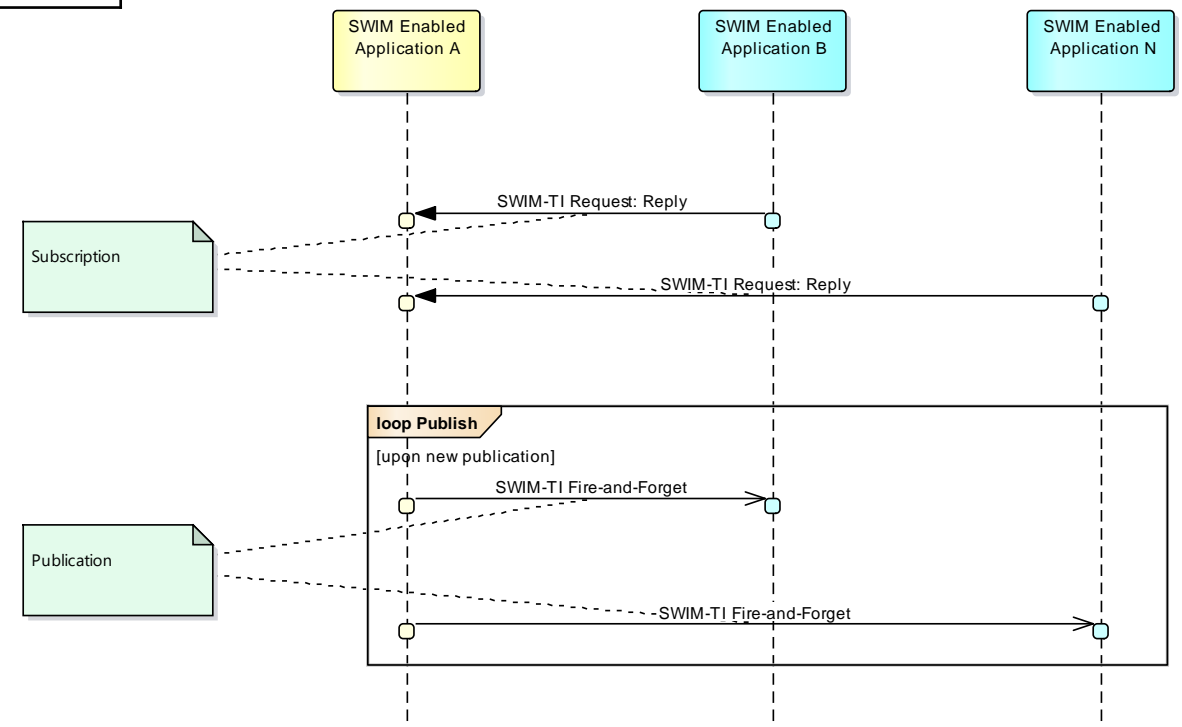
Publish/Subscribe Pull

Cardinality	1-many
Space Decoupling	NO
Time Decoupling	NO
Process Decoupling	Asynchronous/Synchronous



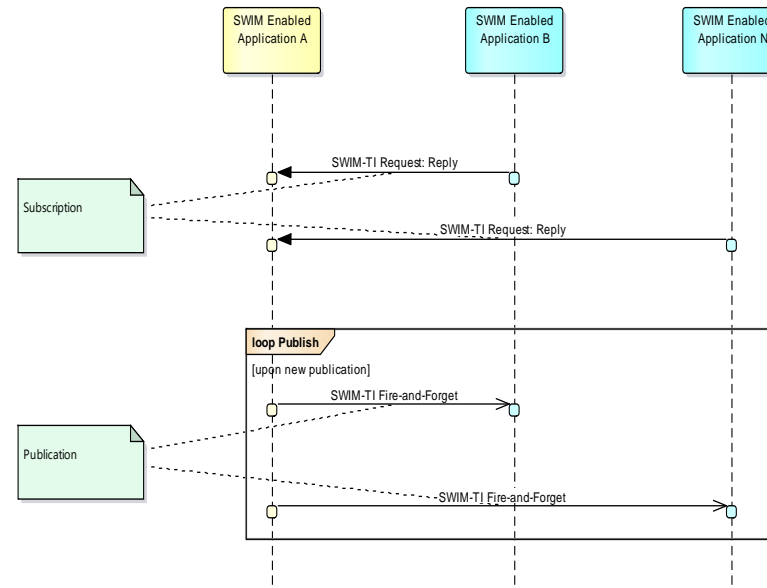
Publish/Subscribe Push

Cardinality	1-many
Space Decoupling	NO
Time Decoupling	NO
Process Decoupling	Asynchronous

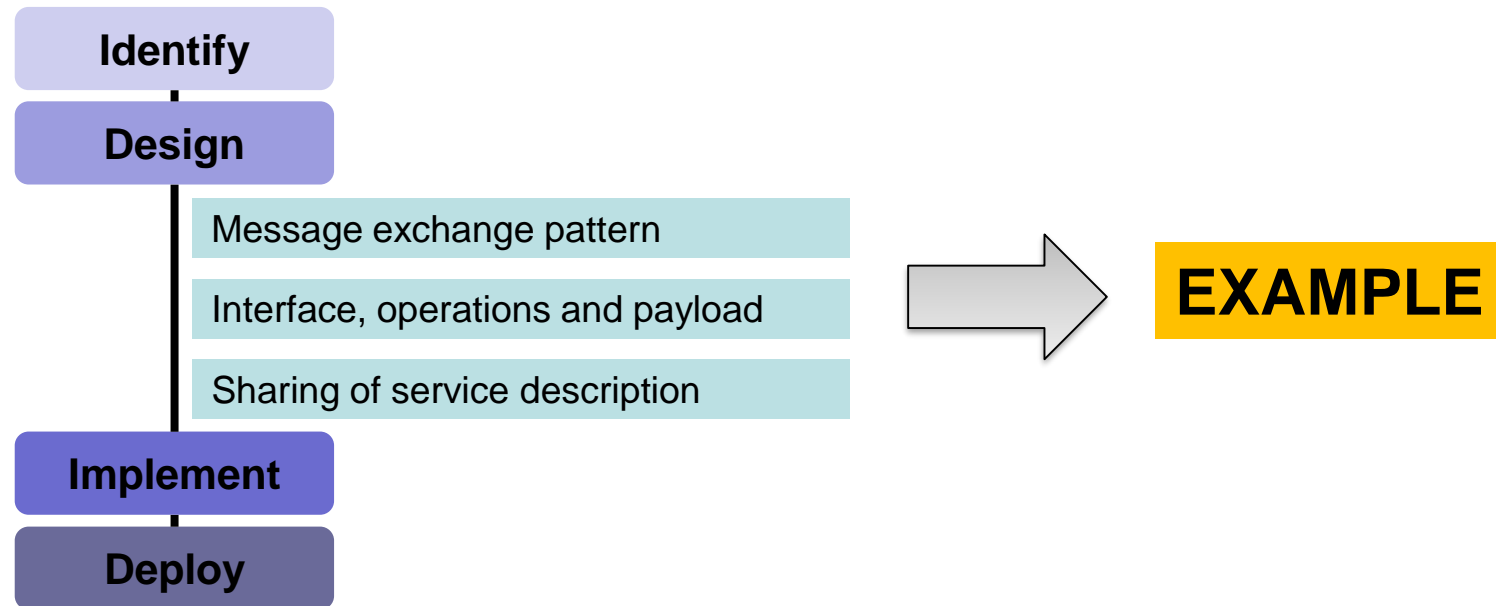


MEP Identification

Pub/Sub Push



Interface Design



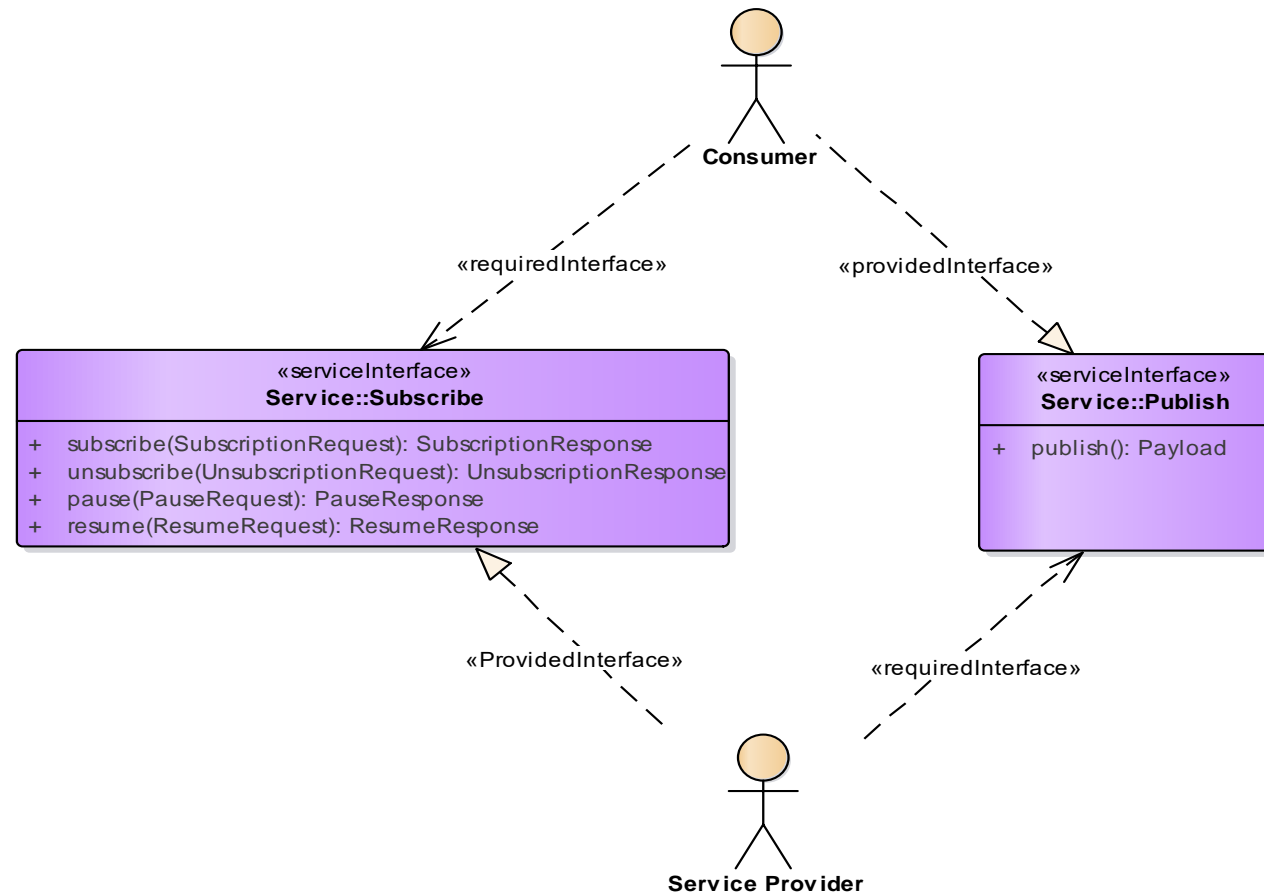
Resources



SWIM TI Guidance for Pub/Sub Push implementation

<https://ost.eurocontrol.int/sites/AISWIM/SWIMspecs/TEC/Shared%20Documents/Supporting%20Material/04%20-%20SWIM%20TI%20Guidance%20for%20Pub-Sub%20Push%20Implementation/SWIM%20TI%20Guidance%20for%20Pub-Sub%20Push%20Implementation.docx>

Pub/Sub Pattern (Abstract) – Interfaces and Operations



Subscription Interface: Operations

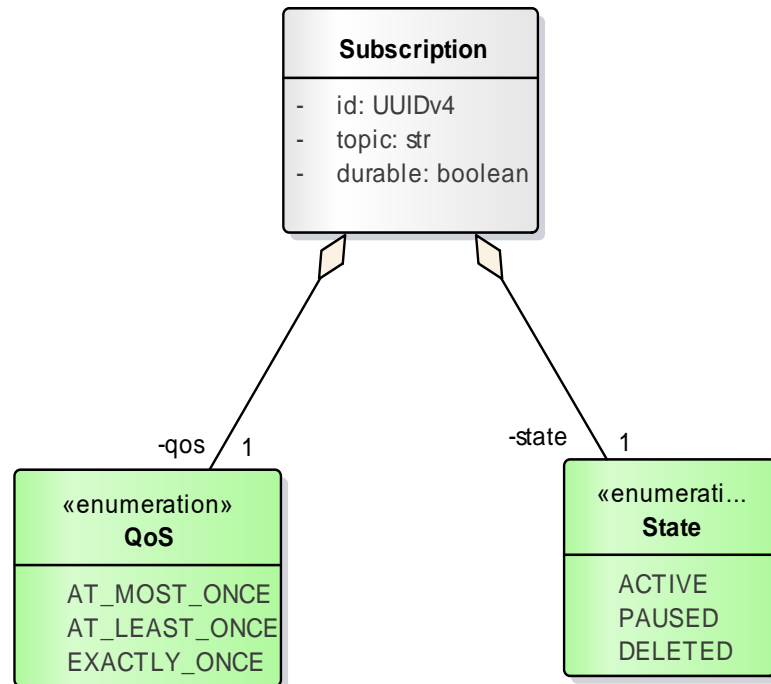
- Operations
 - *Subscribe*
 - *Unsubscribe*
 - *Pause*
 - *Resume*



Subscription Interface

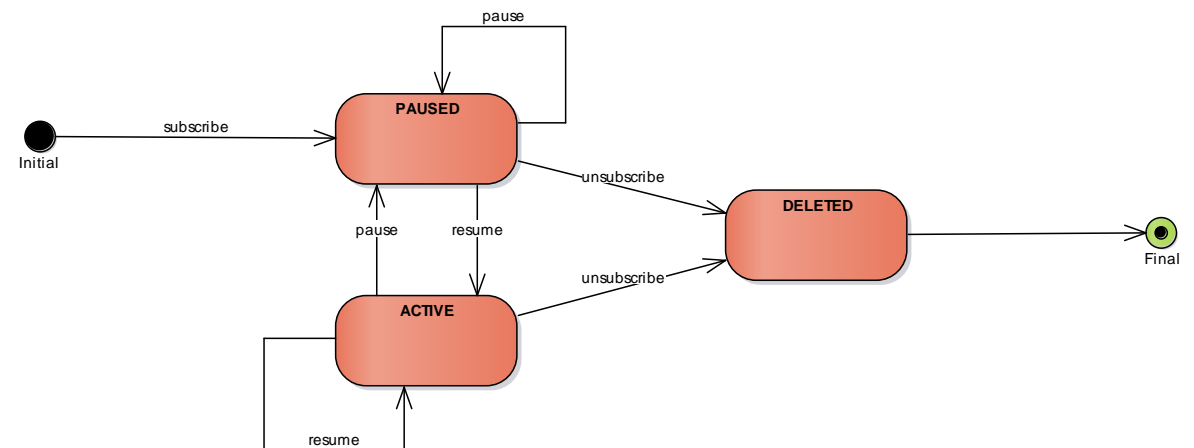
Data Definition

(describing data structure and acceptable values)



Service Behaviour

(describing interface, operation dependencies)



Subscription Interface: Filtering

- Topics
 - Flight Alerts
 - Inbound Flights
 - Outbound Flights

Exchanged information:

Flight Alert: Flight Identifier, Alert Code, Alert Text, Alert Status code, Alert Level

Inbound Flight: Flight Identifier, Aircraft Identification, Aircraft Registration, Aircraft Type code, Departure Aerodrome IATA code, Departure Aerodrome ICAO code, Runway designator, Aircraft Stand designator, SIBT, ELDT, EIBT, ALDT, AIBT, Aircraft Flight Status

Outbound Flight: Flight Identifier, Aircraft Identification, Aircraft Registration, Aircraft Type code, Destination Aerodrome IATA code, Destination Aerodrome, ICAO code, Gate designator, Aircraft Stand designator, Runway designator, SID designator, SOBT, EOBT, TOBT, TOBT update status code, TSAT, TTOT, CTOT, EXOT, ACGT, ASBT, ARDT, ASRT, ASAT, AOBT, ATOT, Aircraft Flight Status



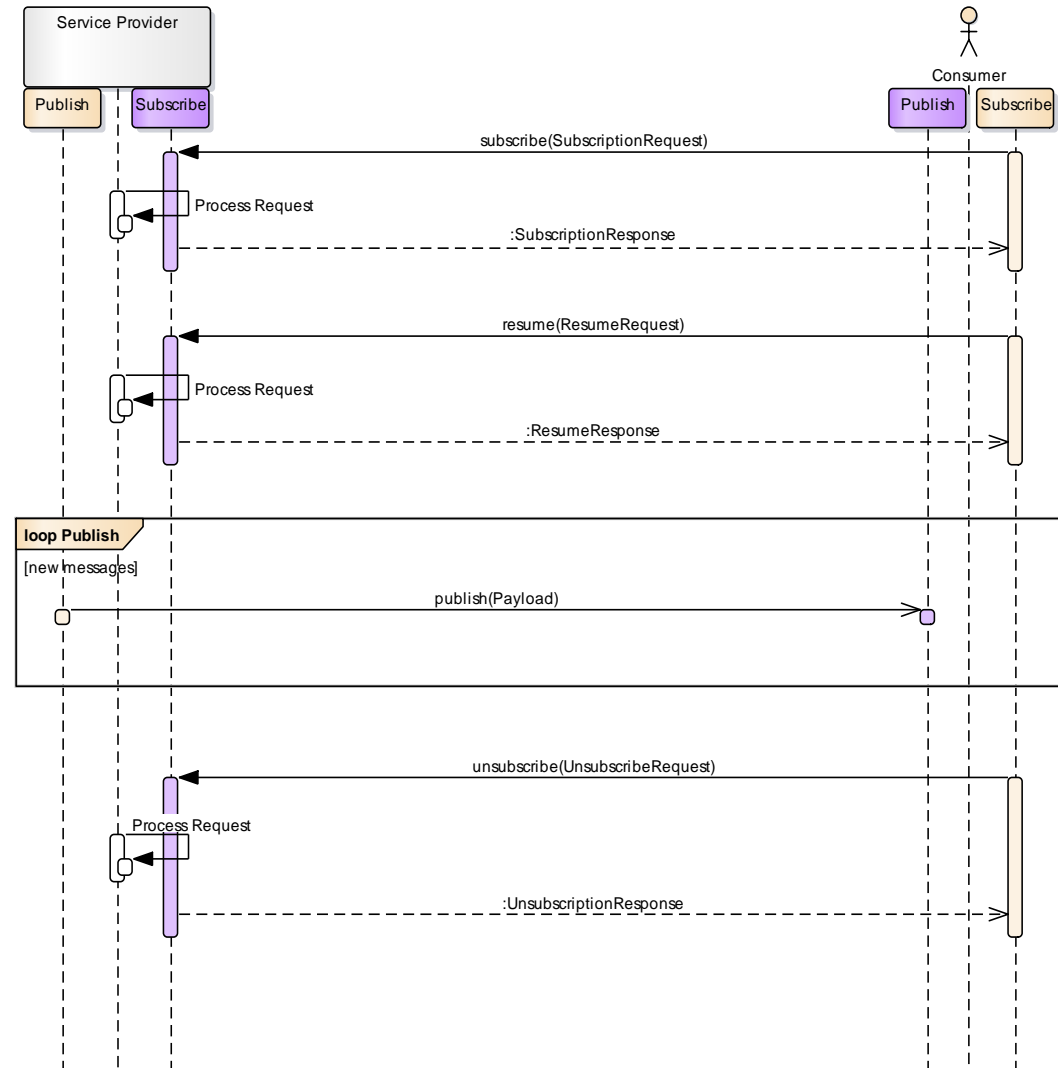
- Subscription Interface
 - Subscribe Operation
 - SubscriptionRequest (Message)

Publish Interface: Operations

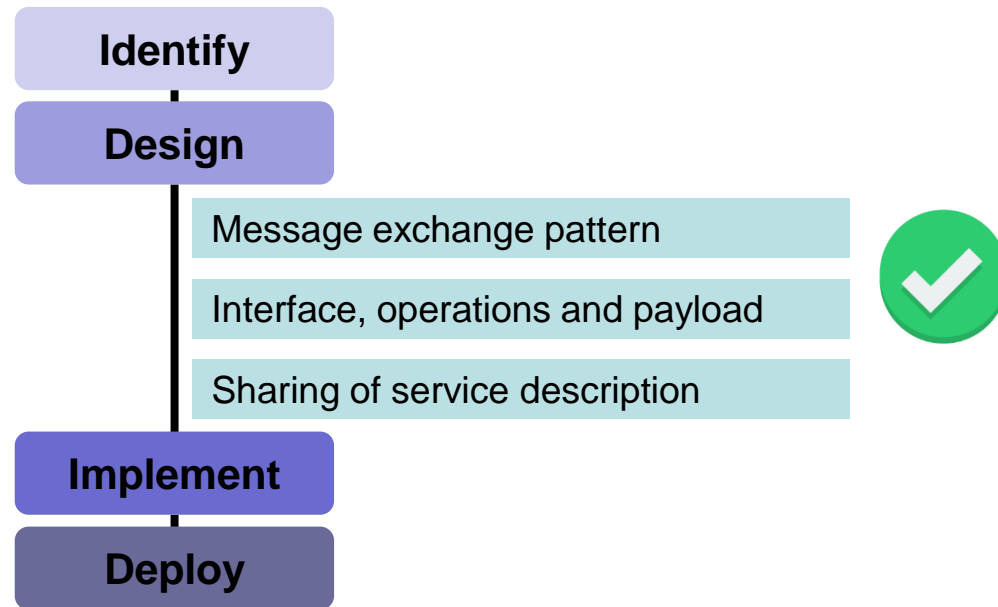
- Operations
 - *Publish*



Service Behaviour




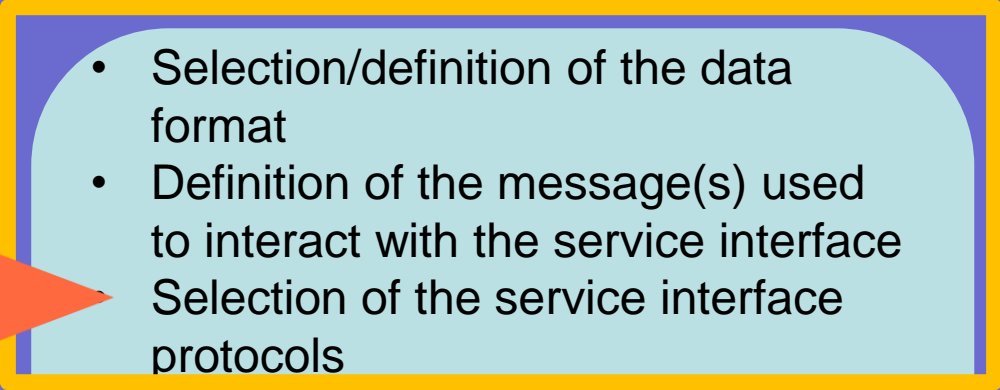
Interface Design




Implement

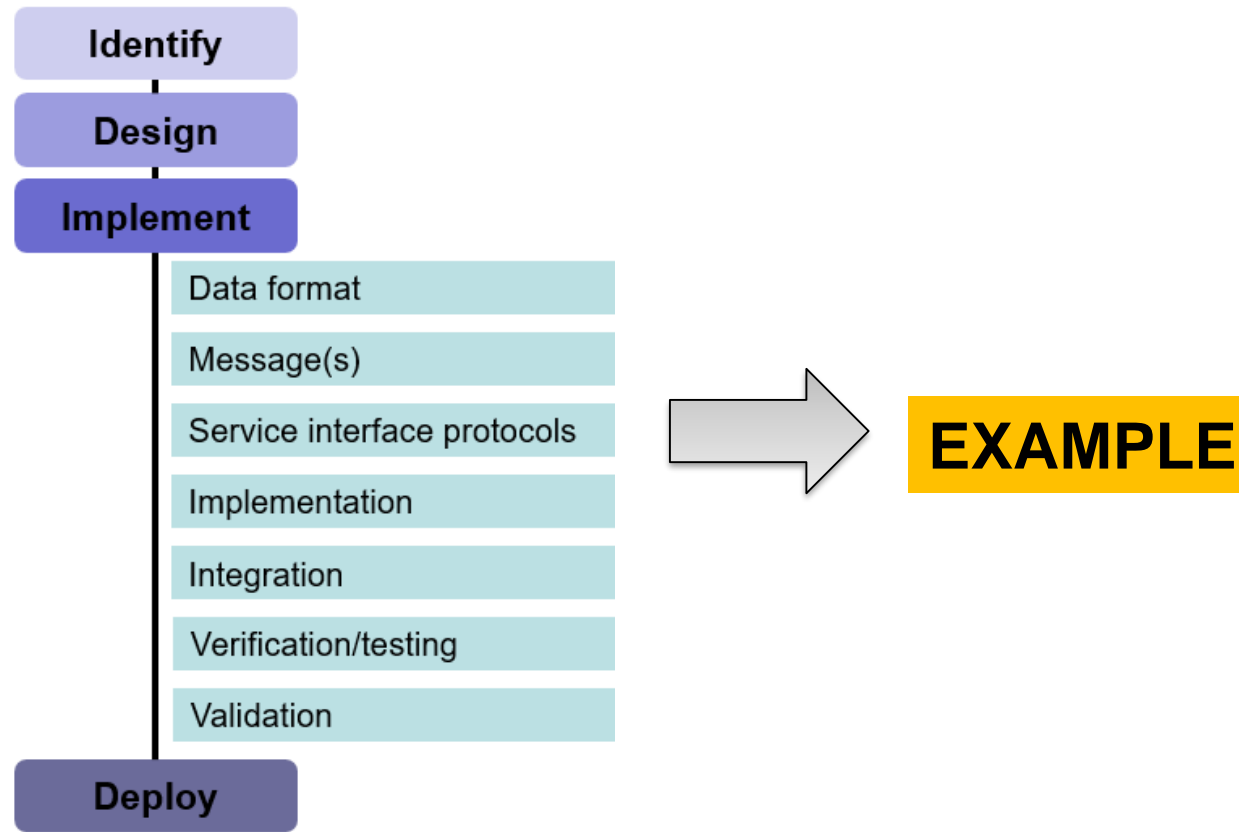
service implementation is the set of activities where the information service is implemented in a target environment and technology context.

Technical Interoperability Design

- 
- 
- Selection/definition of the data format
 - Definition of the message(s) used to interact with the service interface
 - Selection of the service interface protocols
 - Implementation of service(s) using technology and based on implementation choices made
 - Integration of the service(s) into the target environment
 - Verification and testing of the service(s)
 - Validation of the service(s)

- 
- Chosen XM or other data format.
 - Chosen service interface protocol.
 - Message definition
 - Implemented service(s) (interfaces and operations)
 - Machine readable service definition
 - Verification report, validation report
 - Service Overview (update)
 - Verification report
 - Validation report
 - Service implementation information (e.g. service interface protocols and QoS characteristics)

(Subscribe Interface) Interface Binding Determination



Resources

Resources



SWIM TI Binding Selection Guidelines

<https://ost.eurocontrol.int/sites/AISWIM/SWIMspecs/TEC/Shared%20Documents/Supporting%20Material/03%20-%20SWIM%20TI%20Binding%20Selection%20Guidelines/SWIM%20TI%20Binding%20Selection%20Guidelines.docx>

Resources



SWIM TI Guidance for Pub/Sub Push implementation

<https://ost.eurocontrol.int/sites/AISWIM/SWIMspecs/TEC/Shared%20Documents/Supporting%20Material/04%20-%20SWIM%20TI%20Guidance%20for%20Pub-Sub%20Push%20Implementation/SWIM%20TI%20Guidance%20for%20Pub-Sub%20Push%20Implementation.docx>

Binding Selection

- **WS Light**
- **WS SOAP**
- **WS SOAP with Basic Message Security**
- **WS SOAP with Message Security**
- **WS SOAP with Federated Security**
- **WS-N SOAP**
- **WS-N SOAP with Basic Message Security**
- **WS-N SOAP with Message Security**
- **WS-N SOAP with Federated Security**
- **AMQP Messaging**



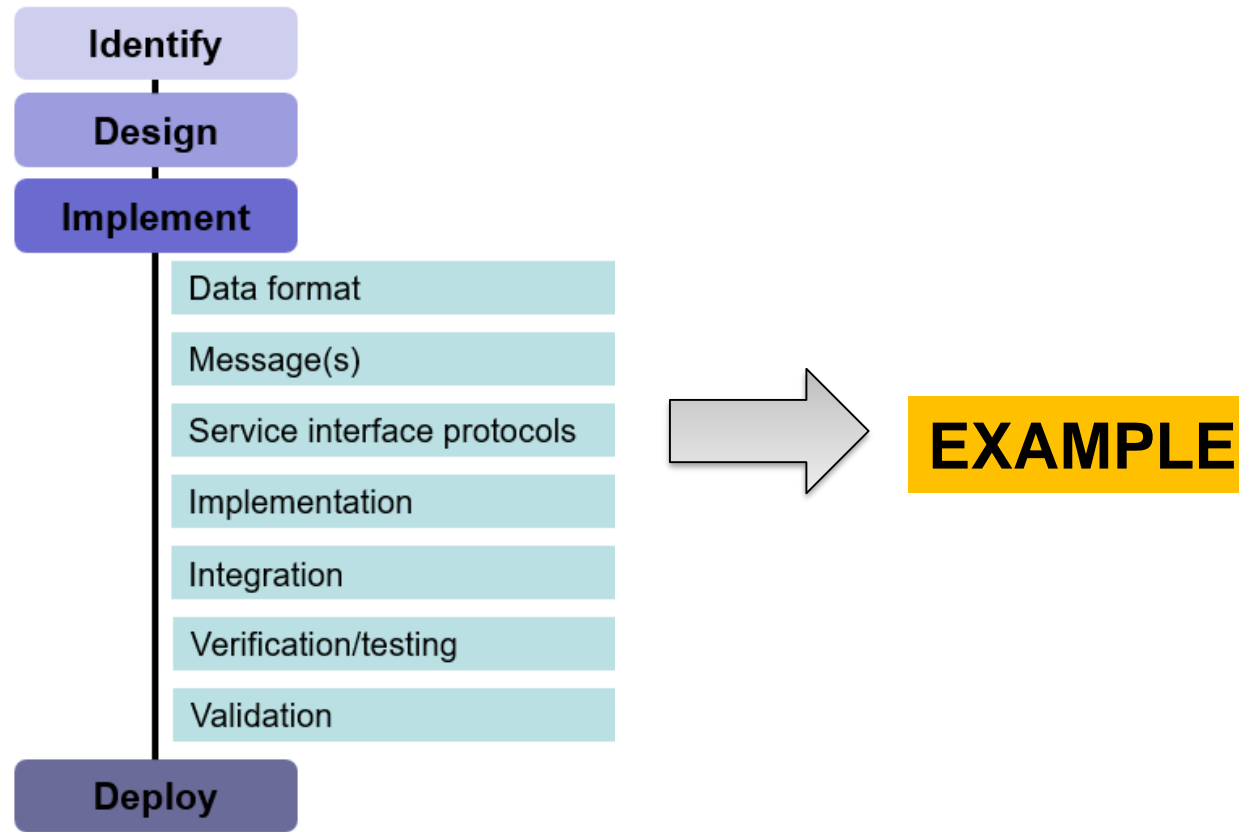
Subscribe interface



Publish interface

- Using WS-Light Service Binding for the implementation of the Subscribe interface,
- Using AMQP Messaging Service Binding for the implementation of the Publish interface.

(Subscribe Interface) Interface Binding Configuration



Binding Configuration

WS-Light

Identifier	Title	Level of Implementation
SWIM-TIYP-0041	Data Compression	O
SWIM-TIYP-0044	HTTP Reason Phrase Header	M
SWIM-TIYP-0043	HTTP Status Code Header	M
SWIM-TIYP-0042	TLS Authentication	M
SWIM-TIYP-0039	HTTP Header Transfer Encoding	O
SWIM-TIYP-0038	HTTP Compression and Content Encoding Header	M Conditional
SWIM-TIYP-0033	HTTP Content Type Header	M Conditional
SWIM-TIYP-0010	HTTP over TLS	M
SWIM-TIYP-0009	HTTP	M
SWIM-TIYP-0008	TLS	M

Binding Configuration

Title	Level of Implementation
Data Compression	O
HTTP Reason Phrase Header	M
HTTP Status Code Header	M
TLS Authentication	M
HTTP Header Transfer Encoding	O
HTTP Compression and Content Encoding Header	M Conditional
HTTP Content Type Header	M Conditional
HTTP over TLS	M
HTTP	M
TLS	M

Data transmission optimization

- ❖ None of the following mechanisms available in the binding will be implemented:
- Compression
 - Chunked transfer encoding

Binding Configuration

Title	Level of Implementation
Data Compression	O
HTTP Reason Phrase Header	M
HTTP Status Code Header	M
TLS Authentication	M
HTTP Header Transfer Encoding	O
HTTP Compression and Content Encoding Header	M Conditional
HTTP Content Type Header	M Conditional
HTTP over TLS	M
HTTP	M
TLS	M



HTTP Implementation

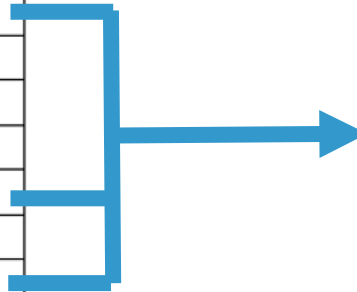
- ✓ HTTP1.1
- ✓ HTTP POST
- ✓ HTTP responses including status code and phrase header.

The services make use of the standard HTTP 400 error [Bad Request] in any of the following cases:

- The request is for an unsupported release
- The request is not a well-formed XML
- The request is a well-formed XML but it is not valid with respect to the XSD (i.e. it does not conform to the type and attribute names defined in the XSD and documented in the reference manuals). Examples of causes for invalid XML documents are:
 - Unexpected element or attribute
 - Element order violation
 - Incorrect primitive value
 - Unexpected enum value

Binding Configuration

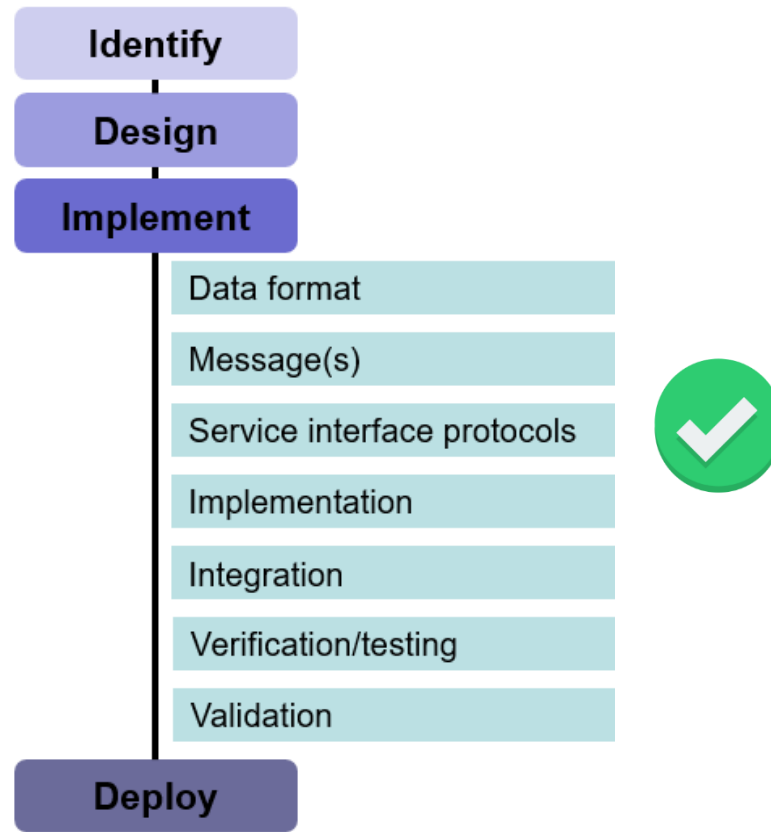
Title	Level of Implementation
Data Compression	O
HTTP Reason Phrase Header	M
HTTP Status Code Header	M
TLS Authentication	M
HTTP Header Transfer Encoding	O
HTTP Compression and Content Encoding Header	M Conditional
HTTP Content Type Header	M Conditional
HTTP over TLS	M
HTTP	M
TLS	M



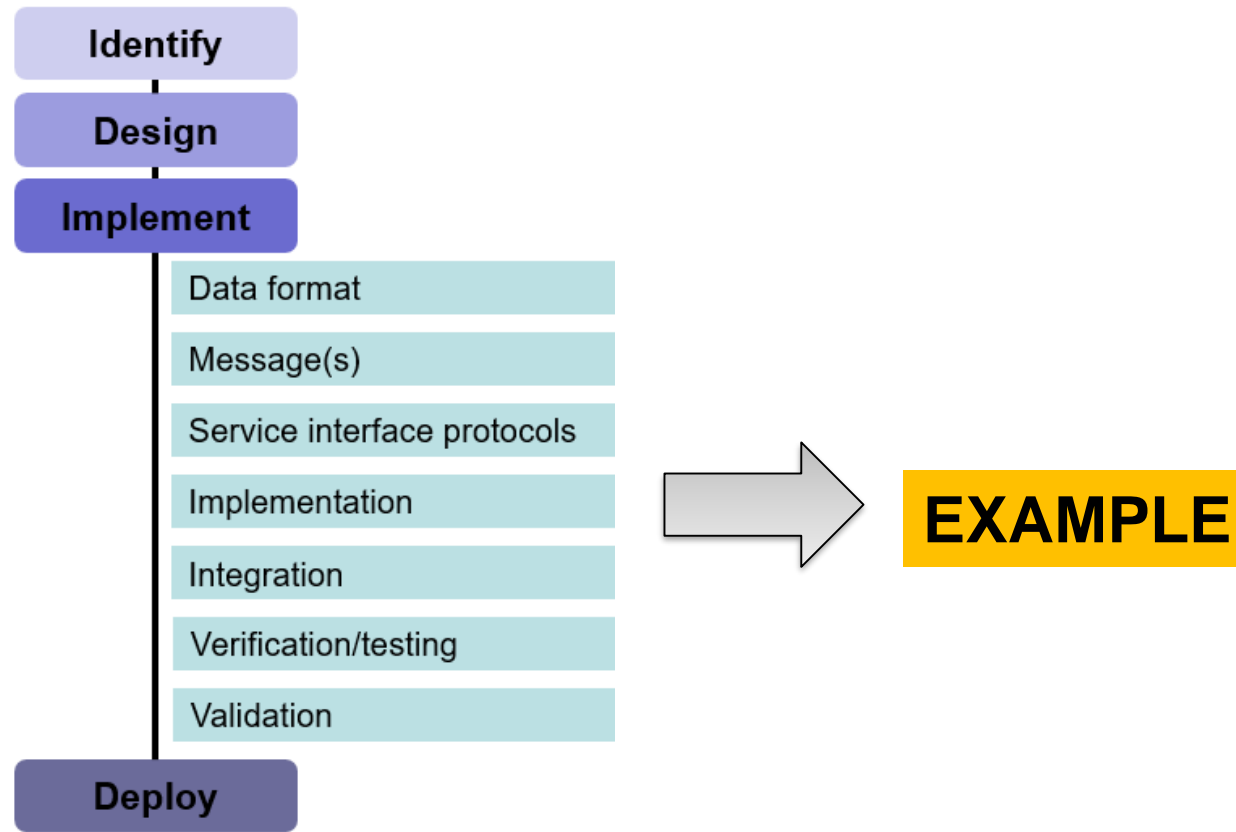
Security protocols

- ✓ TLS1.2
- ✓ Server authentication with X.509 and Client authentication with HTTP Basic or ~~HTTP Digest~~.
- ✓ AES_128_GCM_SHA256, AES_256_CCM

(Subscribe Interface) Interface Binding



(Subscribe Interface) Service Operations Implementation



Subscribe Interface: Operation Implementation

- Operations
 - *subscribe*
 - *unsubscribe*
 - *pause*
 - *resume*



- Transport protocol mapping

Design	Implementation	
Operation	Resource endpoint	HTTP method
subscribe	/subscriptions/	POST
pause	/subscriptions/<subscription_id>	PUT
resume	/subscriptions/<subscription_id>	PUT
unsubscribe	/subscriptions/<subscription_id>	DELETE

Subscription Interface: Operation Implementation

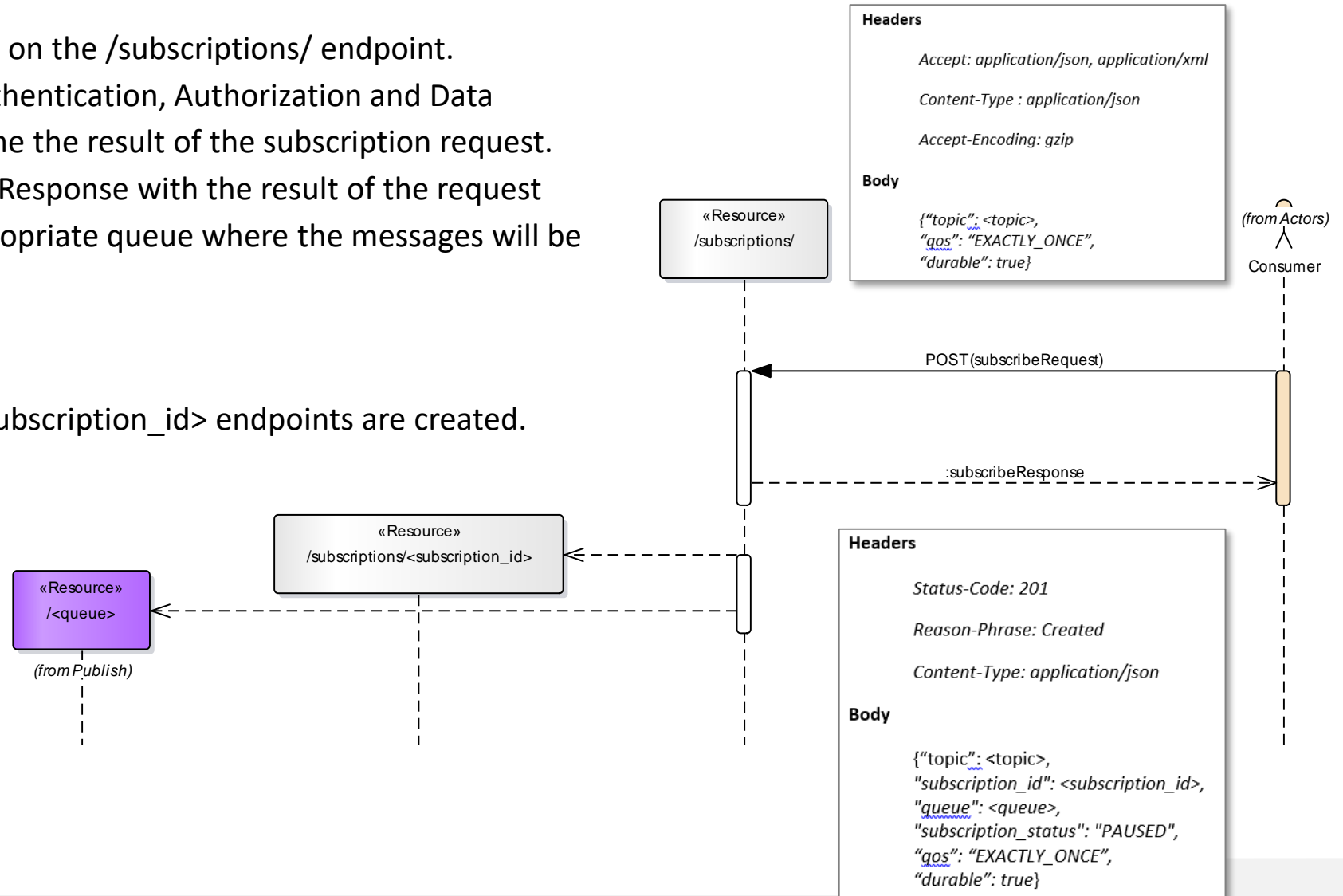
1. The Consumer invokes a “POST” on the /subscriptions/ endpoint.
2. The Service Provider applies Authentication, Authorization and Data Validation controls and determine the result of the subscription request.
3. The Service Provider provides a Response with the result of the request including (if successful) the appropriate queue where the messages will be published.

Post-Conditions: (If successful)

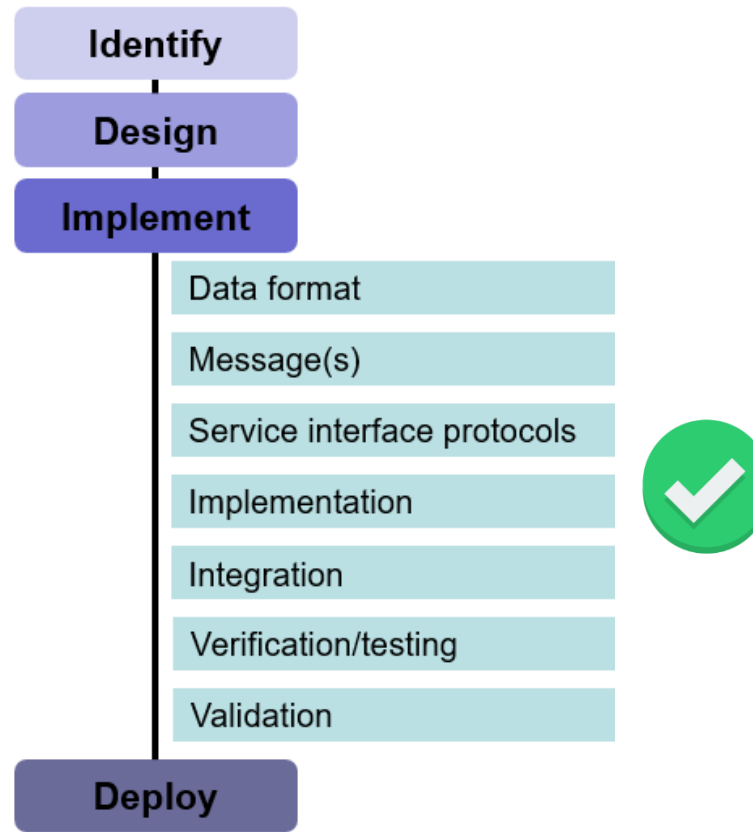
/<queue> and /subscriptions/<subscription_id> endpoints are created.

Error Codes:

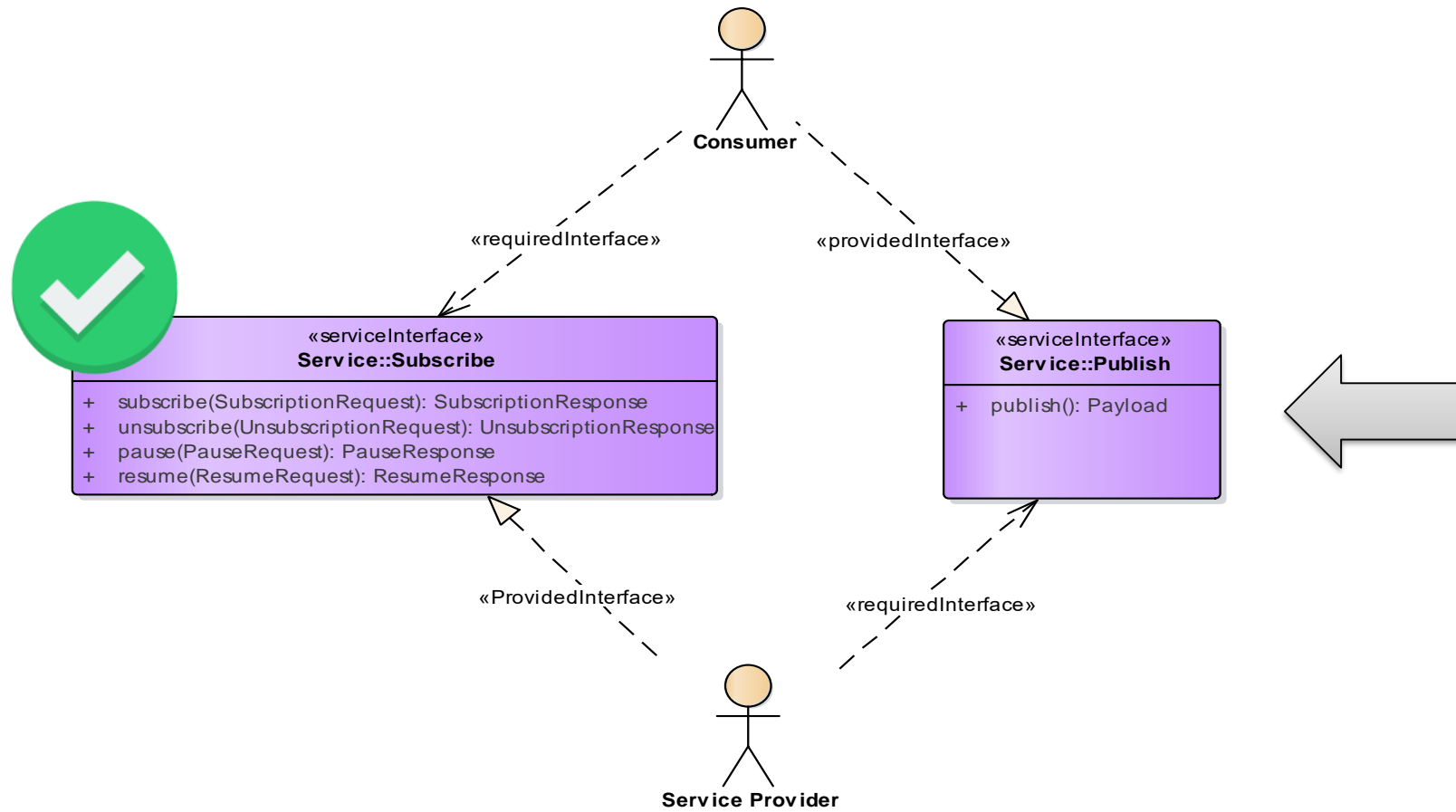
- 400 Bad Request:** Indicated for malformed requests on the part of the client.
- 401 Unauthorized:** Indicated for authentication errors (e.g. the authentication is not provided, the credentials used do not exist, the authentication mechanism used is not valid...).
- 403 Forbidden:** Indicated for authorization errors (e.g. the authenticated credentials are not allowed to perform this operation on the resource).
- 405 Method Not Allowed:** The HTTP method used is not supported by the resource.
- 406 Not Acceptable:** None of the Content-Types provided in the request Accept header are supported.
- 415 Unsupported Media Type:** The Content-Type of the request is not valid for this operation.
- 500 Internal Server Error:** Catch-all error for unexpected internal errors during the processing of the request.



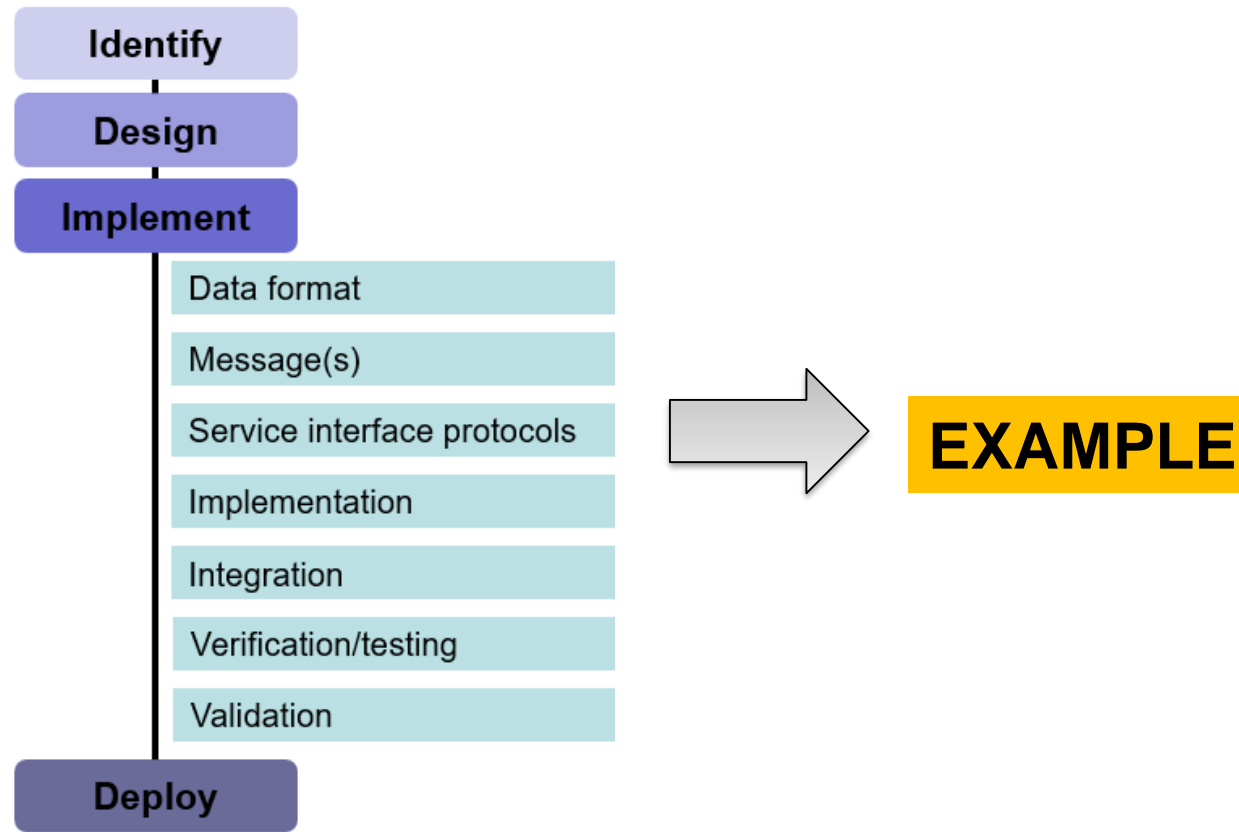
(Subscribe Interface) Service Operation Implementation



Service Interfaces



(Publish Interface) Interface Binding Configuration



Binding Configuration

AMQP-Messaging

Identifier	Title	Level of Implementation
SWIM-TIYP-0041	Data Compression	O
SWIM-TIYP-0049	AMQP Content Encoding Header	O
SWIM-TIYP-0048	AMQP Content Type Header	M Conditional
SWIM-TIYP-0037	AMQP Transport Security Authentication	M
SWIM-TIYP-0051	SASL	M
SWIM-TIYP-0052	AMQP over TLS	M
SWIM-TIYP-0036	AMQP	M
SWIM-TIYP-0008	TLS	M

Binding Configuration

Title	Level of Implementation
Data Compression	O
AMQP Content Encoding Header	O
AMQP Content Type Header	M Conditional
AMQP Transport Security Authentication	M
SASL	M
AMQP over TLS	M
AMQP	M
TLS	M

Data transmission optimization

- ❖ None of the following mechanisms available in the binding will be implemented:
 - Compression

Binding Configuration

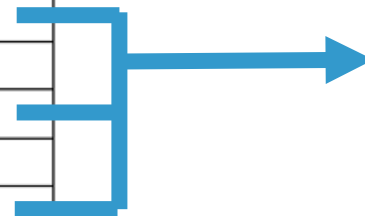
Title	Level of Implementation
Data Compression	O
AMQP Content Encoding Header	O
AMQP Content Type Header	M Conditional
AMQP Transport Security Authentication	M
SASL	M
AMQP over TLS	M
AMQP	M
TLS	M

AMQP Implementation

- ✓ AMQP 1.0
- ✓ Use of content type and encoding headers

Binding Configuration

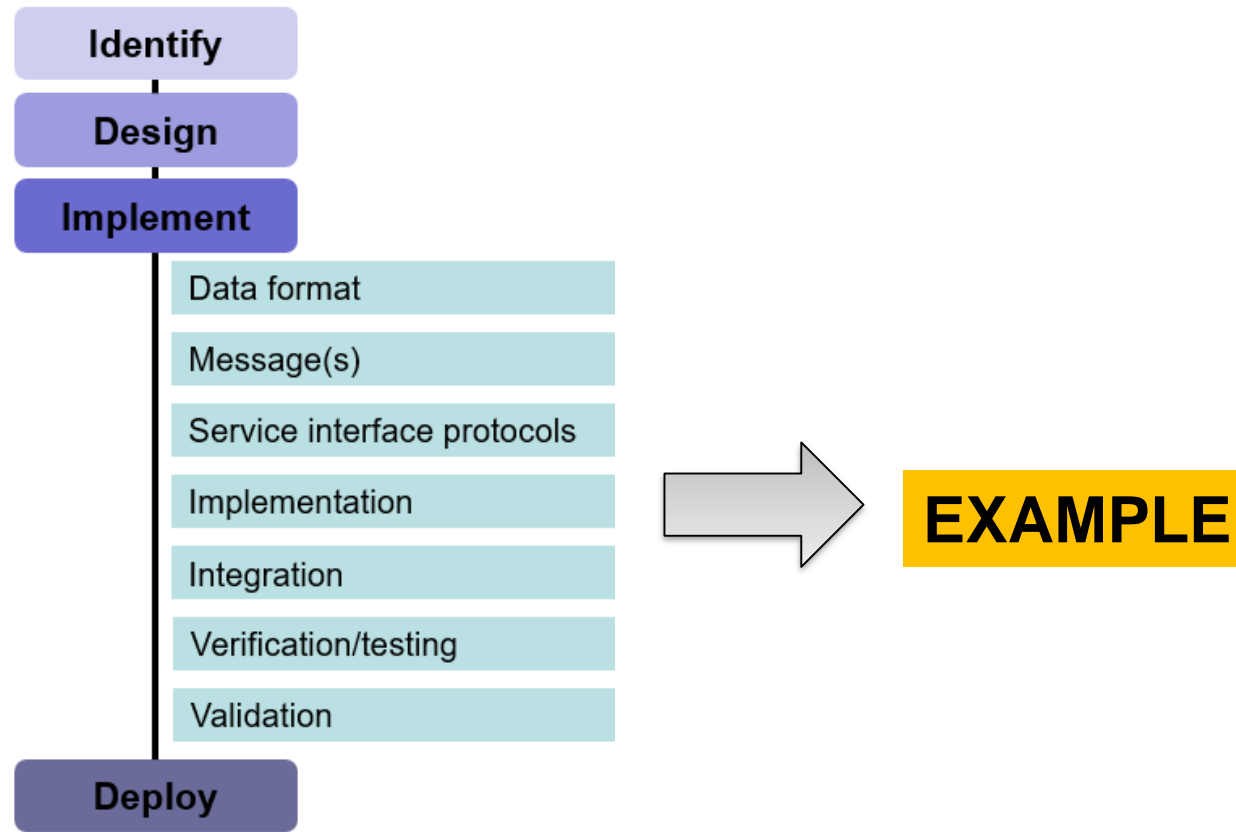
Title	Level of Implementation
Data Compression	O
AMQP Content Encoding Header	O
AMQP Content Type Header	M Conditional
AMQP Transport Security Authentication	M
SASL	M
AMQP over TLS	M
AMQP	M
TLS	M



Security protocols

- ✓ TLS1.2
- ✓ Server authentication with X.509 and Client authentication with SASL uid/pwd.
- ✓ AES_128_GCM_SHA256, AES_256_CCM

(Publish Interface) Service Operations Implementation



Publish Interface: Operation Implementation

- Operations
 - Publish*



Design	Implementation	
Operation	Resource endpoint	AMQP 1.0 method
publish	/<queue>	transfer



Publish Interface: Operation Implementation

Pre-conditions:

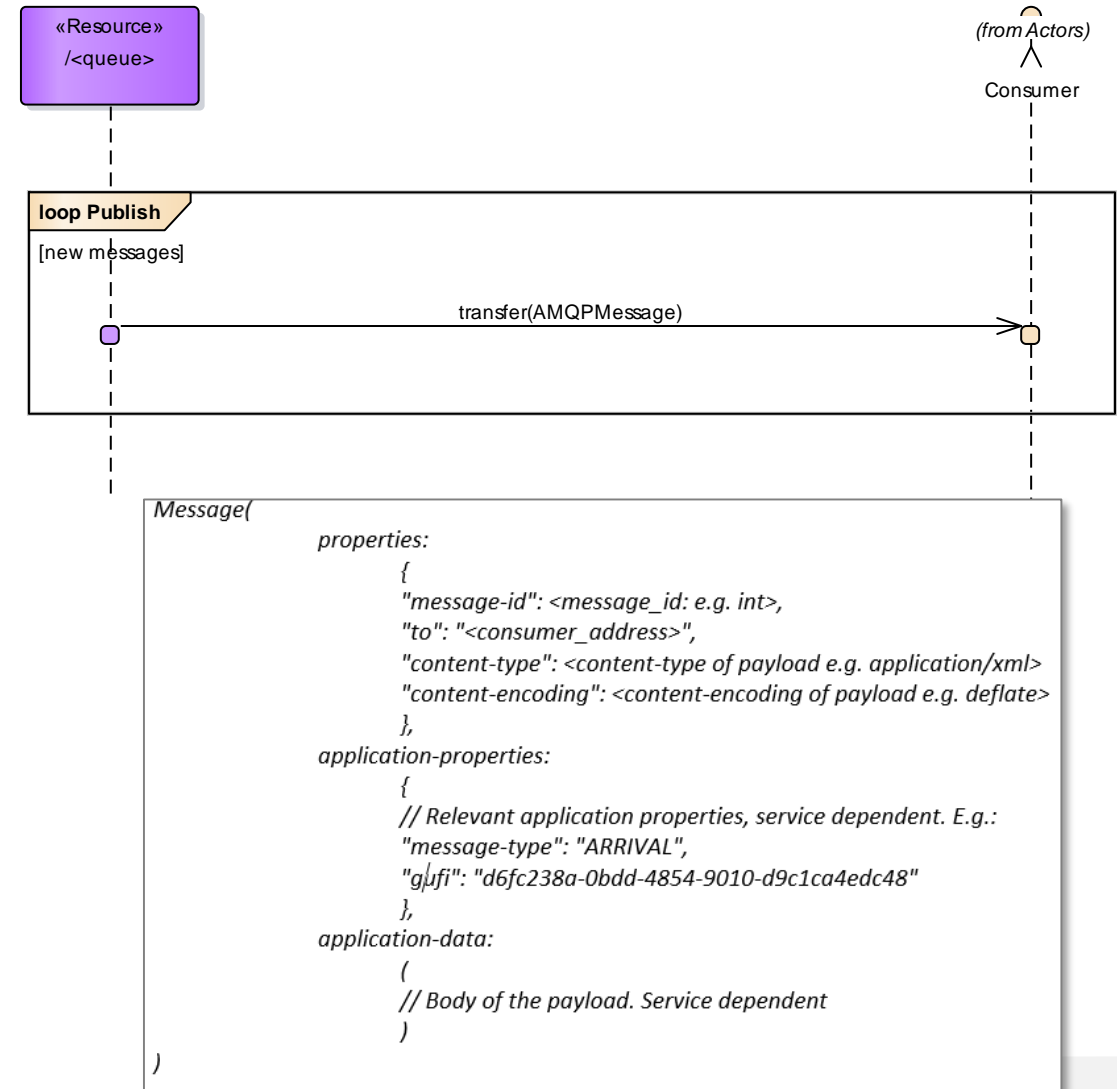
- The Consumer has subscribed to a <topic> and connected to the <queue> provided in the subscribe response message.
- There is sufficient link-credit in the consumer's link endpoint.

Steps

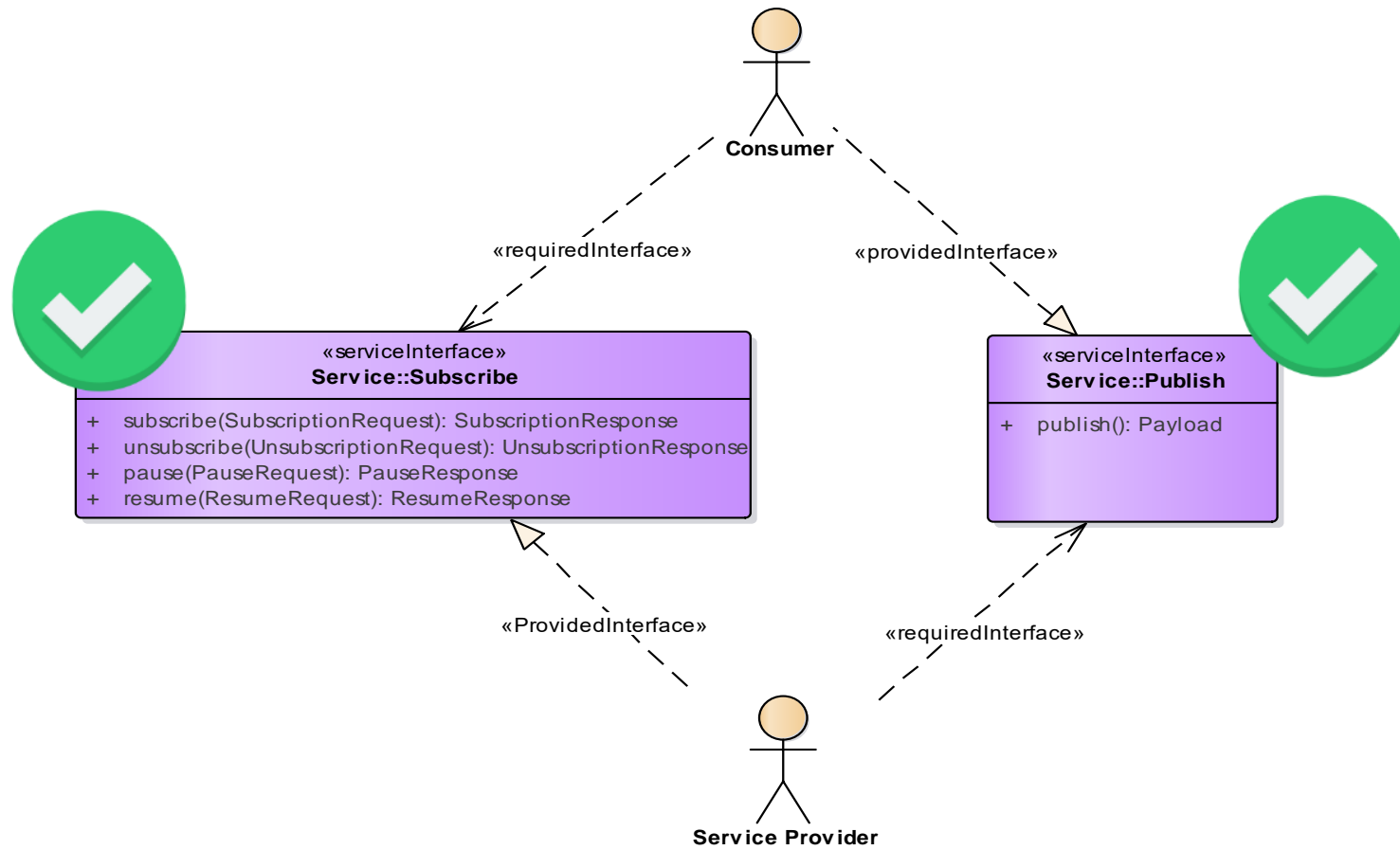
1. The Provider sends a message with the appropriate payload using the "transfer" performative of AMQP 1.0.

Post-Conditions:

- If successful, the message is received.



Service Interfaces



Resources

Resources



SWIM TI Binding Selection Guidelines

<https://ost.eurocontrol.int/sites/AISWIM/SWIMspecs/TEC/Shared%20Documents/Supporting%20Material/03%20-%20SWIM%20TI%20Binding%20Selection%20Guidelines/SWIM%20TI%20Binding%20Selection%20Guidelines.docx>

Resources



SWIM TI Guidance for Pub/Sub Push implementation

<https://ost.eurocontrol.int/sites/AISWIM/SWIMspecs/TEC/Shared%20Documents/Supporting%20Material/04%20-%20SWIM%20TI%20Guidance%20for%20Pub-Sub%20Push%20Implementation/SWIM%20TI%20Guidance%20for%20Pub-Sub%20Push%20Implementation.docx>