Web Service Reliability

NOKIA

Content

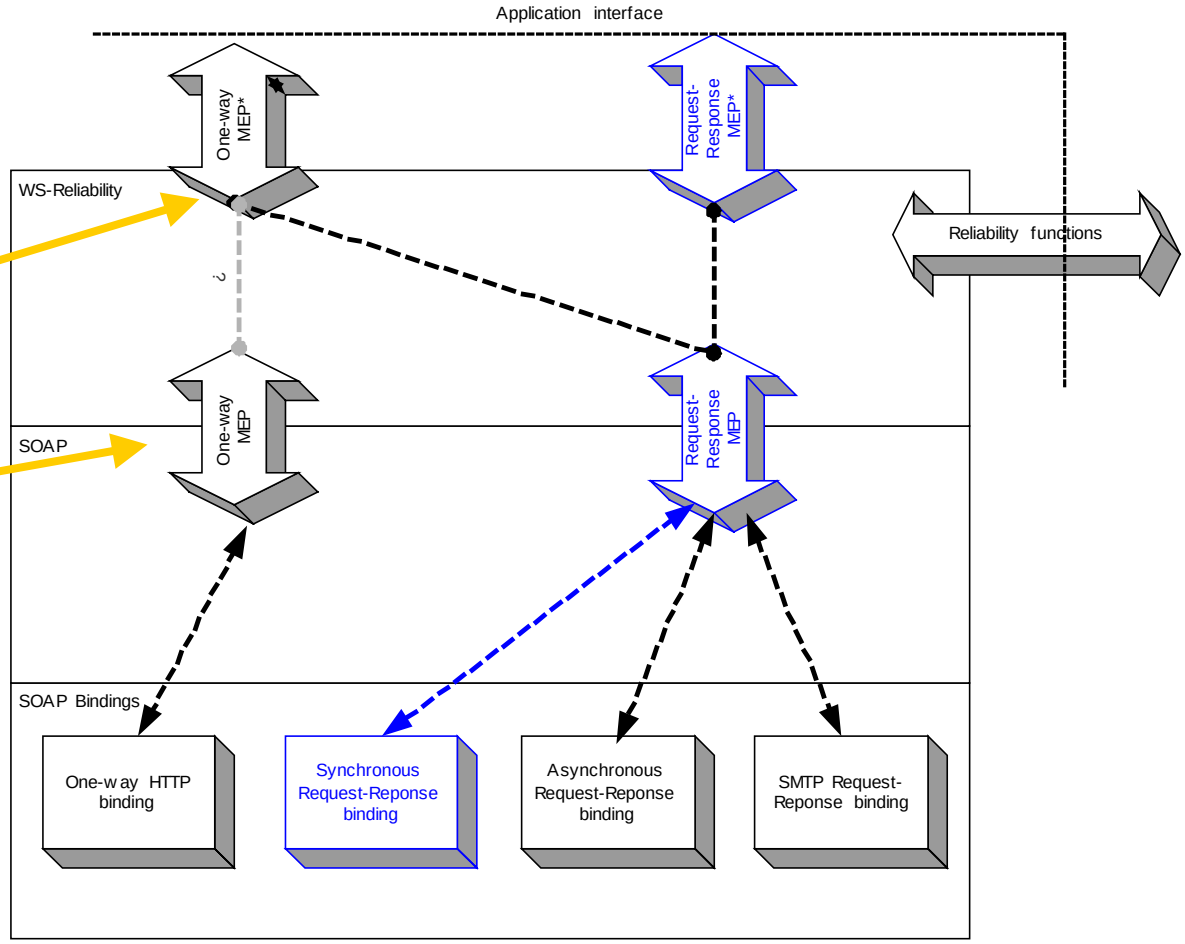
- What is reliability ?
  - Guaranteed Delivery
  - Duplicate Elimination
  - Ordering
  - Crash tolerance
  - State synchronization
- Reliability aspects
- Business use cases to be supported
- SOAP Message Exchange Patterns
  - One-way MEP pattern
  - Request-Response MEP
- Requirements
- Solution proposal

NOKIA

What is Web Service Reliability?

- Web Service Reliability is a **communication layer** in a Web Services protocol stack.

Ws-Reliability offers Reliable SOAP MEPs upwards, and uses Standard SOAP MEPs downwards



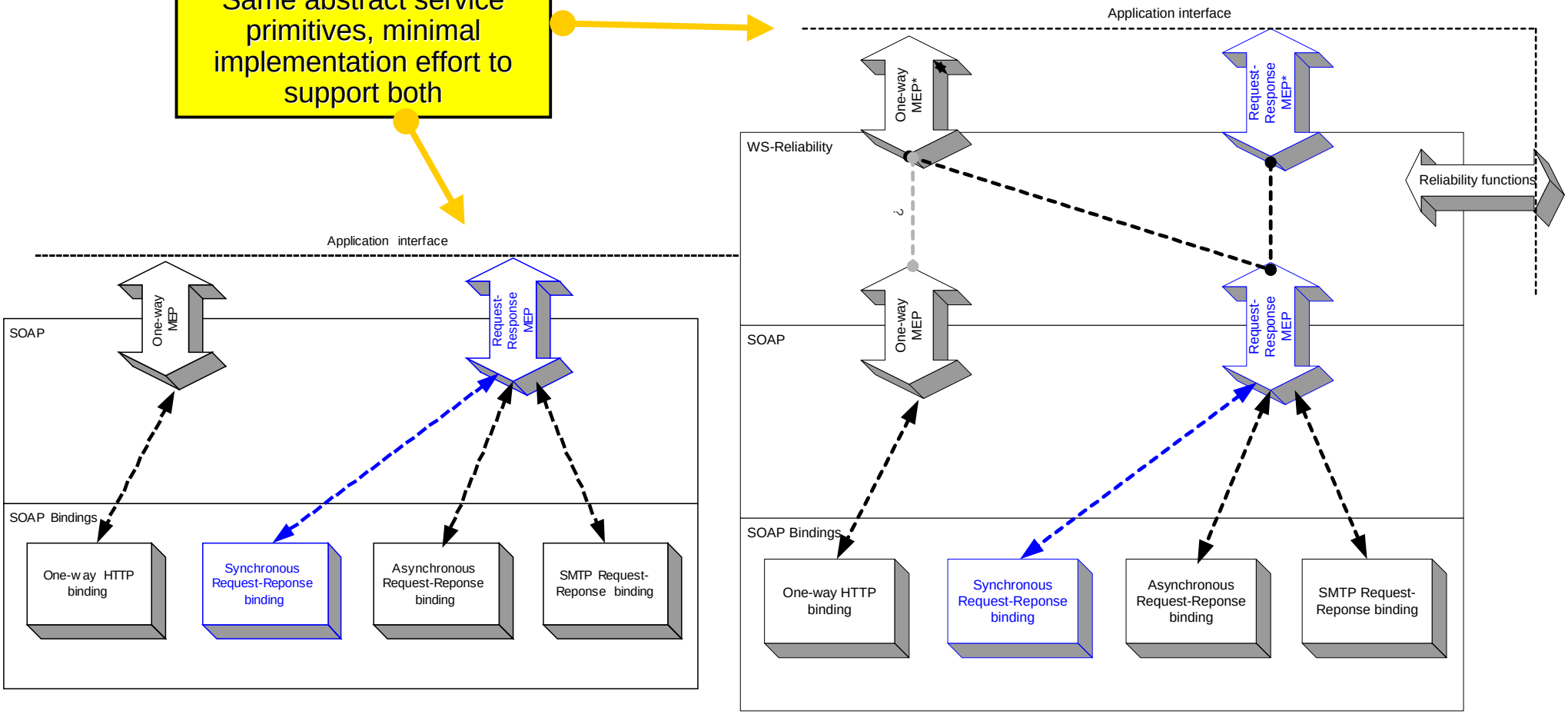
Blue = standard SOAP

Extension of existing SOAP

Standard SOAP

SOAP with WS-Reliability

Same abstract service primitives, minimal implementation effort to support both



Reliability aspects

- **Guaranteed delivery**: ensure that all information to be sent actually received by the destination or error reported.
- **Duplicate Elimination**: ensure that all duplicated information can be detected and filtered out.
- **Ordering**: communication between parties consist of several individual Message Exchanges. This aspect ensures that Message Exchanges are forwarded to the receiver application in the same order as the sender application issued.
- **Crash tolerance**: ensures that all information prescribed by the protocol is always available regardless of possible physical machine failure.
- **State synchronization:** If the MEP is cancelled for any reason then it is desirable for both nodes to set their state as if there were no communication between the parties.

NOKIA

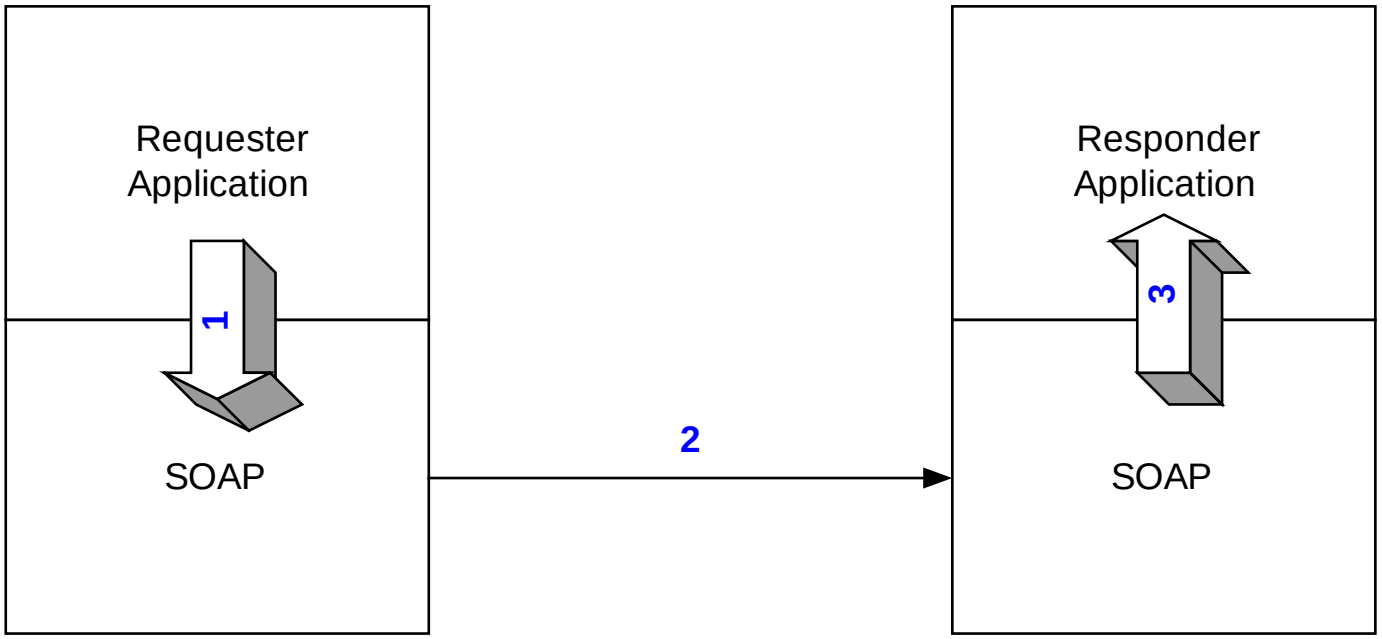
Business use cases for the same service (MMS)

- Advertisement company wants to send bulk MMS ads to its 15000 registered customers.
  - Duplicate Elimination is a nice-to-have (no disturbing multiple ads)
  - Guaranteed Delivery and Ordering not needed (cost-effectiveness is more important)
  - Crash tolerance not seen important
- Advertisement company wants to send customized ad MMS to one of its customers
  - Duplicate Elimination is a nice-to-have (no disturbing multiple ads)
  - Guaranteed Delivery needed (cost of customization should be guaranteed)
  - Ordering not needed
  - Crash tolerance is less important than price of service
- Mobile payment company sends payment receipt in MMS to customer
  - Duplicate Elimination is important
  - Guaranteed Delivery needed
  - Ordering not needed
  - Crash tolerance is important

NOKIA

SOAP One-Way Message Exchange Pattern

- It is a new MEP, as it is not defined in SOAP specification.
- Must be defined to cover original Sun use cases.



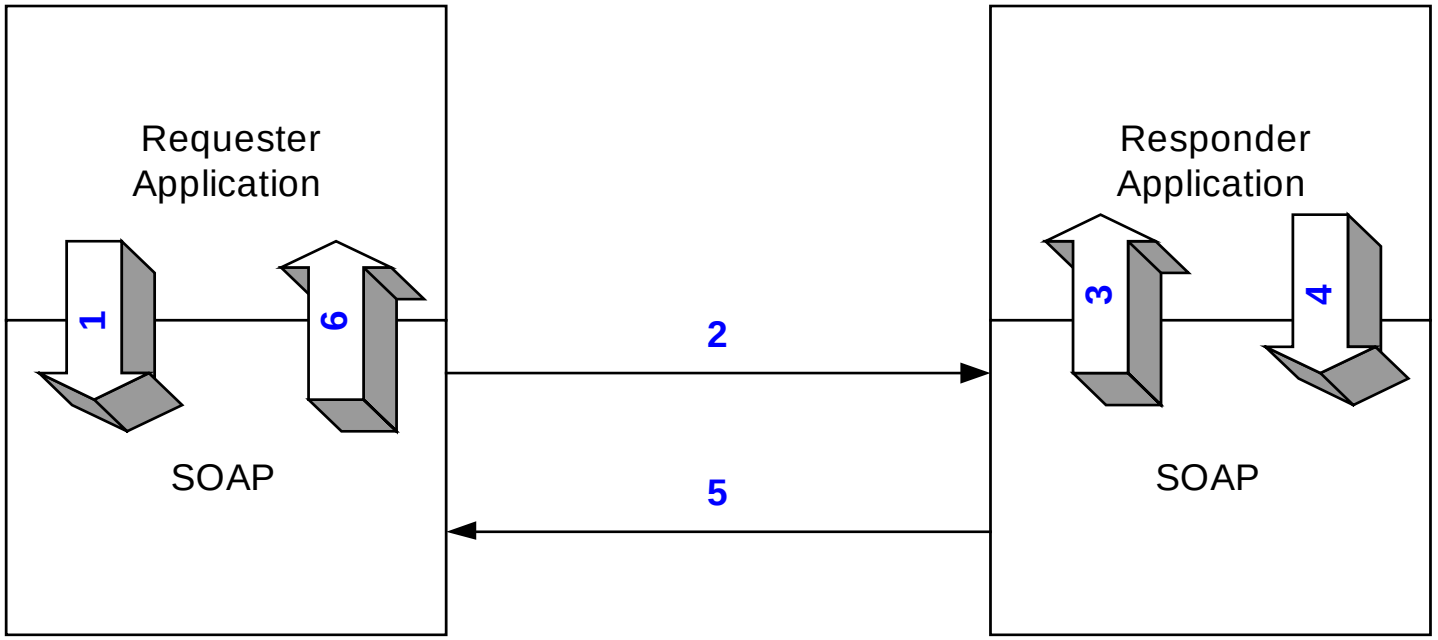
1: SOAP message initiated trough the API
2: SOAP message sent on-the-wire using the actual transport binding
3: Responder Application notified about the incoming message

Guaranteed delivery for One-Way MEP

- Must be solved if SOAP Transport **binding** is not guaranteeing delivery

- From Requester point of view
  After step 1 either
  - Message delivered or
  - Error reported

SOAP Request-Response Message Exchange Pattern



1: SOAP request initiated trough the API
2: SOAP request sent on-the-wire using the actual transport binding
3: Responder Application notified about the incoming request
4: Responder application answers
5: SOAP response sent on-the-wire using the actual transport binding
6: Requester application notified about the answer

NOKIA

Guaranteed delivery for Request-Response MEP

- Must be solved if SOAP Transport **binding** is not guaranteeing delivery (This is the case with the standard HTTP binding)
- From Requester point of view
  After step 1 either
  - Step 6 will occur at some time (Message Exchange closed) or
  - Error reported
- From Responder point of view
  After step 4 either
  - Step 6 will occur at some time (Message Exchange closed) or
  - Error reported

- Both must be satisfied
- Because Responder depends on Step 6, MEP must be extended for delivering information to Responder after step 4.

NOKIA

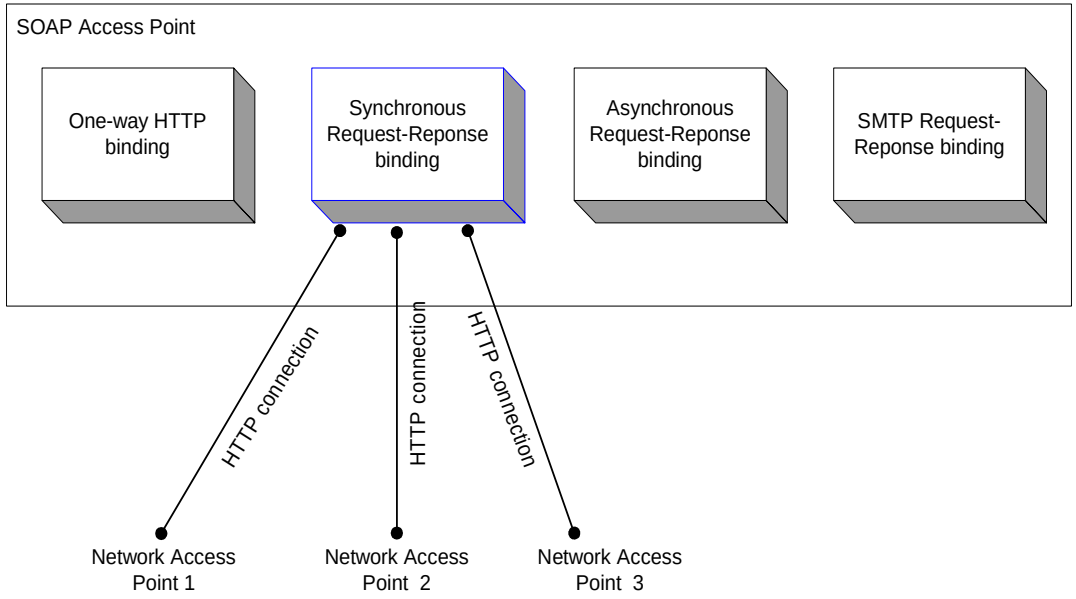
Duplicate elimination (for all MEPs)

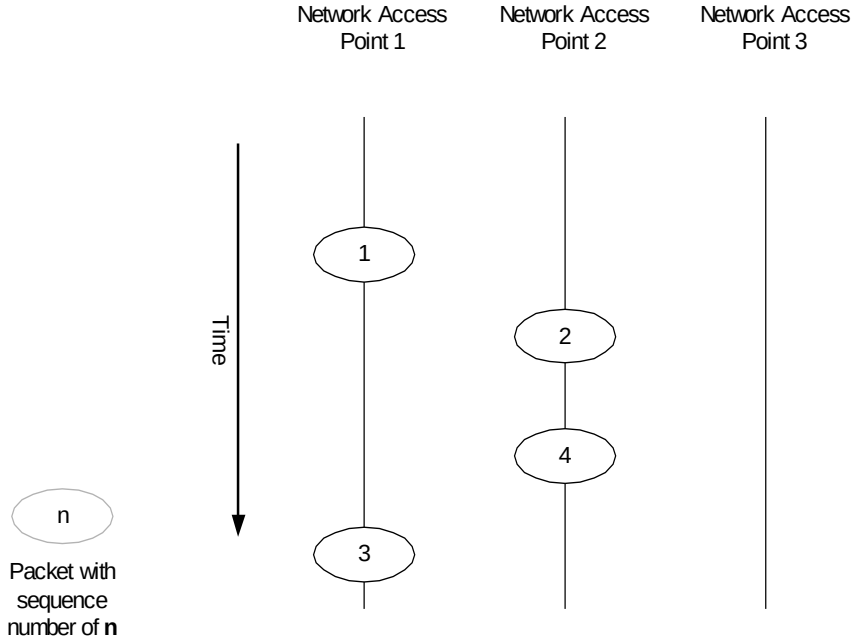
- Must be solved if transport binding doesn’t offer duplicate elimination
- Duplicated messages must not be received by the application

Ordering (for all MEPs)

- Ordering of Message Exchange Patters must be solved if
  - Transport binding doesn't ensure ordering of messages

    or
  - Multiple network access points are used between two communication parties (see figures below)
- Be aware, that using Ordering means a kind of session !





Crash tolerance (for all MEPs)

- Crash tolerance should be an optional feature with more levels

- No persistent storage
  - lost message content without specific indication of crash
  - possible replay
- Persistent storage of MEP metadata
  - lost message content, but specific indication of crash
  - MEP replay is not possible
- Persistent storage of MEP metadata and content
  - No lost content
  - No replay

NOKIA

State synchronization (for all MEPs)

- There are cases when MEP is broken despite of any effort (for example network cable is cut)

- The problem can be solved by persistent storage of message contents for unlimited time
- If this is not possible then consistent states on both ends can be ensured by rollback on the Responder side
  - If Rollback is available, then in case of broken MEP, the state of Responder must be set back to pre-MEP state.
- If not possible then application level action is needed to synchronize states on both ends (!)

Requirements

- Maximal reuse of existing implementations
  - Interoperability with SOAP nodes not supporting the reliability feature
    - » Fallback to non-supporting mode
    - » Indication by the source if fallback is acceptable
  - API offered by Reliable SOAP should be an extension of what classic SOAP offers
  - SOAP Message Exchange Patterns should be supported
  - SOAP Transport bindings should be supported
- Persistent storage must not be mandated
  - It must not be indicated as a mandatory feature in the specification to store messages in a persistent storage.
- Levels of reliability should be defined
  - Choose **duplicate elimination** or **reliable message delivery** as possible functionalities. **Message ordering** is not a high priority for us, it should be optional. Strict s**tate synchronization** should be optional.
  - Minimal levels of **persistent storage** usage:
    - » Persistent storage of all message content
    - » Persistent storage of MEP metadata
    - » No persistent storage of any data
- SOAP **intermediaries** should be supported

NOKIA

Solution

- Definitions:
  - *Standard Request-Response MEP* is the MEP defined by SOAP 1.2 Part 2 that is equivalent with the SOAP 1.1 HTTP binding.
  - Here we denote the content of the Standard Request-Response MEP as a *Standard Request* and a *Standard Response*.

- Solution proposal:

Only for consistency

  - Define a (very simple) state machine for the One-Way MEP (according to SOAP 1.2 Part 2 Section 6)
  - The HTTP binding for One-Way MEP is already defined in WS-I Basic Profile
  - Define two **state-machines:**
    - Reliable One-way MEP using Standard Request-Response MEP
    - Reliable Request-Response using Standard Request-Response MEP

(All details are not covered here)

NOKIA

Reliable One-way MEP

- Consist of a Reliable Message abstract service primitive

- Needs two transport-level, logical messages:
  - **Req:** A message conveying the content of the Reliable Message. (Requester -> Responder)
  - **Ack:** A message containing an acknowledgement (Responder -> Requester)

- The logical messages are bound to the Standard Request-Response MEP the following way:
  - **Req** is conveyed in the Standard SOAP Request
    - There is a mandatory SOAP header indicating that this is a Reliable One-Way MEP
  - **Ack** is conveyed in the Standard SOAP Response
    - <SOAP:Body> is empty, <Ack> header entry added.

NOKIA

Reliable Request-Response MEP

- Consists of a *Reliable Request* and a *Reliable Response,* otherwise the state machine is equivalent with Standard Request-Response MEP's state machine
- Needs three transport-level logical messages:
  - **Req** (Requester -> Responder) containing the Reliable Request
  - **Rsp** (Responder -> Requester) containing the Reliable Response and means implicit acknowledgement of **Req**
  - **Ack** (Requester -> Responder) is the explicit acknowledgement of **Rsp** and closure of MEP
- The logical messages are bound to the Standard Request-Response MEP the following way:
  - **Req** is conveyed in the Standard SOAP Request
    - There is a mandatory SOAP header indicating that this is a Reliable Request-Response MEP
  - **Rsp** is conveyed in the Standard SOAP Response
    - There is an optional SOAP header indicating that reliable messaging is accepted.
  - **Ack** can be conveyed in
    - In an optional <Ack> header entry of a subsequent MEP's **Req**
    - In an <Ack> header of a Standard SOAP Request with empty <SOAP:Body>

NOKIA