|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Message Exchange Pattern** | **Error Handling** | **WSDL 2.0 Pattern** | **WSDL 2.0**  **Fault Type** | **WSDL 2.0 Message Type**  **In the WSDL 2.0 Operation** | **Corresponding BPMN 2.0**  **Element and Attribute** |
| Fire-and-Forget   * Initiator initiates upon activation * Initiated waits upon initiation | No | In-Only | No Faults | **input** *messageLabel*="In" *element*=”{message name}”  “ | **Send Task** or **Throw Type Message Event** as initiator of service – *messageRef* refers (by semantic id) to **message** *name* = “{message name}”  **Receive Task** or **Catch Type Message Event** as initiated service – *messageRef* refers (by semantic id) to **message** *name* = “{message name}” |
| Fire-and-Forget   * Initiator initiates upon activation * Initiated waits upon initiation   How does initiator handle returned error? | Yes | Robust In-Only | Message Triggers Fault | **input** *messageLabel*="In" *element*=”{message name}”  “  **outfault** messageLabel="Out" *ref*=”{message name}” | **Send Task** or **Throw Type Message Event** as initiator of service – *messageRef* refers (by semantic id) to **message** *name* = “{message name}”  **Receive Task** or **Catch Type Message Event** as initiated service – *messageRef* refers (by semantic id) to **message** *name* = “{message name}”  What to do with the error message, other than to model for receipt of the error message, but then what about the **Error Element**? |
| Synchronous Request/Response   * Initiator blocks until response is received * Initiated waits upon initiation, and then responds   How does initiator handle returned error? | Yes | In-Out | Fault Replaces Message | **input** *messageLabel*="In" *element*=”{message name}”  **output** *messageLabel*="In" *element*=”{message name}”  Where does this go, and is the same schema used? | **Service Task** as initiator of service – *inMessageRef* refers (by semantic id) to **message** *name*=”{message name}”  **Service Task** as initiator of service – *outMessageRef* refers (by semantic id) to **message** *name*=”{message name}”  What to do with the error message, other than to model for receipt of the error message, but then what about the **Error Element**? |
| Synchronous Request/Response   * Initiator blocks until response is received * Initiated waits upon initiation, and then responds   How does initiator handle returned error? | Yes | Robust In-Out | Message Triggers Fault | **input** *messageLabel*="In" *element*=”{message name}”  **output** *messageLabel*="Out" *element*=”{message name}”  **outfault** messageLabel="Out" *ref*=”{message name}” | **Service Task** as initiator of service – *inMessageRef* refers (by semantic id) to **message** *name*=”{message name}”  **Service Task** as initiator of service – *outMessageRef* refers (by semantic id) to **message** *name*=”{message name}”  What to do with the error message, other than to model for receipt of the error message, but then what about the **Error Element**? |