



# **ACORD SOAP Messaging Service Overview**

**ACORD/LOMA Forum  
May 2004  
Serge Cayron -  
ACORD**



# Agenda

---

- **Project background**
- **Architectural overview**
- **Document Repository Interface**
- **Implementation cases**

# Project background

---

- **June 2002: call for a proof of concept of SOAP, with existing Payload, based on eMerge framework results**
- **Formed a Reinsurance pilot group of 11 organizations (1 cedent, 3 brokers, 2 reinsurers, 3 exchanges, 2 solution providers)**
- **Worked on a SOAP POC implementation guide**
  - Incorporating the requirements defined by the cross-constituency eMerge Framework group, to be applicable to the three standards
- **Made an implementation pilot in Feb – May 2003**
- **Created ACORD Messaging Service - SOAP Implementation Guide v1.0**
- **Approved by RLC vote in June and released in September 2003**

# Business goals

---

- **Integrated Messaging Service solution, which would allow business messages to be transported together with unstructured electronic documents**
- **Integration of repositories for unstructured supporting documents**

# « Framework » scope

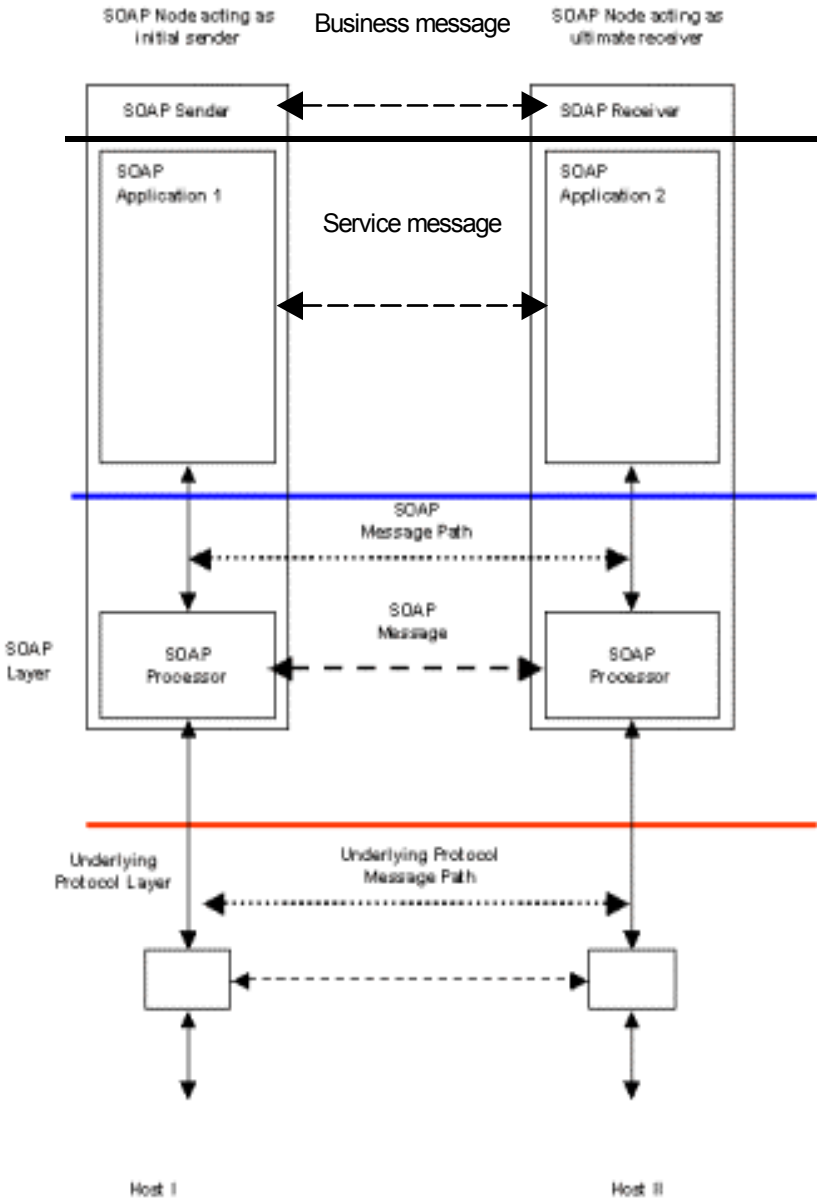
<b>T R A N S P O R T</b>	<b>M E S S A G I N G</b>	<b>T R A N S A C T I O N</b>	<b>P A Y L O A D</b>
<b>SECURITY</b>			
<b>DESCRIPTION AND PUBLICATION</b>			

# Messaging Service - Functional requirements

---

- End to end Addressing
- Packaging
- Synchronous and asynchronous business message exchange
  - Correlation of business requests and responses
- Reliable delivery & delivery in sequence

# Layering



# Design approach

---

- **Part 1: Formal Messaging Service specification**
  - Normative schema for Messaging Operations
  - Strict messaging scenarios
- **Part 2: SOAP implementation conventions**

# Messaging Service

---

- **ACORD XML specification**
- **3 modular protocol mechanisms, described as 3 Web Service « ports ».**
- **Synchronous business message exchange**
  - Call (client calls server application and gets application response)
- **One way – Queuing mode**
  - Inbox (client pushes one way into server inbox queue)
  - Outbox (client pulls one way from server outbox queue)
- **Modular combination of these service ports**

# Messaging Service Operations

---

## ■ Inbox port

- **Post** - push single message from client to server
- **StatusIn** - request status of messages at the server side
- **Listin** - control reliable and sequenced transfer service

## ■ Outbox port

- **Retrieve** - retrieve single message from server to client
- **StatusOut** - report status of retrieved messages to server
- **ListOut** - control reliable and sequenced transfer service

## ■ Call port

- **Call** - send single message as a request for immediate response by the processing application

## ■ Operations work as request/response message pairs (e.g. PostRq/PostRs)

# Operation message - PostRq

The screenshot displays the XMLSpy interface for a SOAP message titled 'ac:PostRq'. The message structure is as follows:

- ac:PostRq**
  - ac:Sender**
    - ac:PartyId: urn:duns:123456789
    - ac:PartyRoleCd: broker
  - ac:Receiver**
    - ac:PartyId: urn:duns:912345678
    - ac:PartyRoleCd: reinsurer
  - ac:Application**
    - ac:ApplicationCd: Jv-Ins-Reinsurance
    - ac:SchemaVersion: <http://www.ACORD.org/Standards/Jv-Ins-Reinsurance/2003-1>
  - ac:TimeStamp: 2003-01-18T13:10:00
  - ac:MsgItem**
    - ac:MsgId: f81d4fae-7dec-11d0-a765-00a0c91e6bf9
    - ac:MsgTypeCd: Placing
  - ac:WorkFolder**

The software interface includes a menu bar (File, Edit, Project, XML, DTD/Schema, Schema design, XSL, Authentic, Convert, View, Browser, Tools, Window, Help), a toolbar, and a status bar at the bottom showing 'XMLSPY v5 rel. 3 UI' and system icons.

# Operation message – PostRs

The screenshot displays the XMLSpy interface with a SOAP message structure for an `ac:PostRs` operation. The message is organized into several sections:

- ac:Sender**
  - `ac:PartyId`: urn:duns:912345678
  - `ac:PartyRoleCd`: reinsurer
- ac:Receiver** (highlighted in blue)
  - `ac:PartyId`: urn:duns:123456789
  - `ac:PartyRoleCd`: broker
- ac:Application**
  - `ac:ApplicationCd`: Jv-Ins-Reinsurance
  - `ac:SchemaVersion`: <http://www.ACORD.org/Standards/Jv-Ins-Reinsurance/2003-1>
- ac:TimeStamp**: 2003-01-18T13:10:00
- ac:MsgItem**
  - `ac:MsgId`: f81d4fae-7dec-11d0-a765-00a0c91e6bf9
  - `ac:MsgTypeCd`: Placing
  - `ac:MsgStatusCd`: received

The software interface includes a menu bar (File, Edit, Project, XML, QTD, Schema, Schema design, XSL, Authentic, Convert, View, Browser, Tools, Window, Help), a toolbar, and a status bar at the bottom showing the version (XMLSPY v5 rel. 3 U) and registration information (Registered to Serge Cayron (ACORD Corporation) ©1998-2003 Altova GmbH & Altova, Inc.). The Windows taskbar at the bottom shows the Start button and several open applications.

# Definitions - Packaging

---

- **Business document** – self-contained piece of aggregated information for business processing
  - **Business message** - standardized document that drives the business process
  - **Attachment** - supporting document associated to a business process but existing separately from business message
- **Work Folder** - set of documents, to be processed within a single unit of work and exchanged as an integral document set.
- **Message** – container of a set of business documents exchanged as a single unit between two parties through the framework.

# WorkFolder

The screenshot displays the XMLSpy interface showing a SOAP message structure. The main window is titled "XMLSPY - [SOAP-Sample-WorkFolder-RLC \*]". The message structure is as follows:

- ac:WorkFolder
  - ac:MsgFile
    - ac:FileId: cid:A01EFAE7-5490-43D0-AB6B-DAEF1671CDCC
    - ac:FileFormatCd: text/xml
  - ac:AttachmentPackages
    - ac:AttachmentPackage
      - ac:Owner
      - ac:CommunicationChannelCd: in\_stream
      - ac:FileId: cid:E2A071D1-CB03-49B5-A58F-144036BF0B84
      - ac:FileFormatCd: application/zip
      - ac:FileSize
      - ac:Attachments
        - ac:Attachment
          - ac:DocumentId: f81d4fae-7dec-11d0-a765-00a0c
          - ac:FileId: 036492871022314.pdf
          - ac:FileFormatCd: application/pdf
          - ac:FileSize
          - ac:DocumentTypeCd: **wording**
          - ac:DocumentReference: -
          - ac:DocumentVersion: -
          - ac:DocumentVersionDtTime: 2003-01-31T00:00:00
          - ac:Description: -

The status bar at the bottom of the window shows "XMLSPY v5 rel. 3 U Registered to Serge Cayron (ACORD Corporation) ©1998-2003 Altova GmbH & Altova, Inc." and the system tray shows the date and time as 20:42.

# SOAP implementation

---

- HTTP binding
- SOAP message packaging
- SOAP Body Namespaces
- SOAP Fault usage
- Direct mapping to WSDL concepts
- Use of main WS-I interoperability recommendations (WS-I Basic profile 1.0)

# HTTP Binding

---

- **Operation requests map to HTTP Requests (POST)**
  - Standard values for SOAPAction attribute
- **Operation responses map to HTTP Responses**
  - Use of HTTP response status as per WS-I basic profile recommendations

# SOAP Message Packaging

## “SOAP Message

### Root Body Mime

#### SOAP Envelope

##### SOAP Header

Optional standard extensions

##### SOAP Body

<Messaging service operation>

<WorkFolder <uuid>Document 987654

<WorkFolder <uuid> Document 12345

### Mime Part

Acord XML message

### Mime Part

Document 12345

### Mime Part

Document 987654

## Standards leveraged

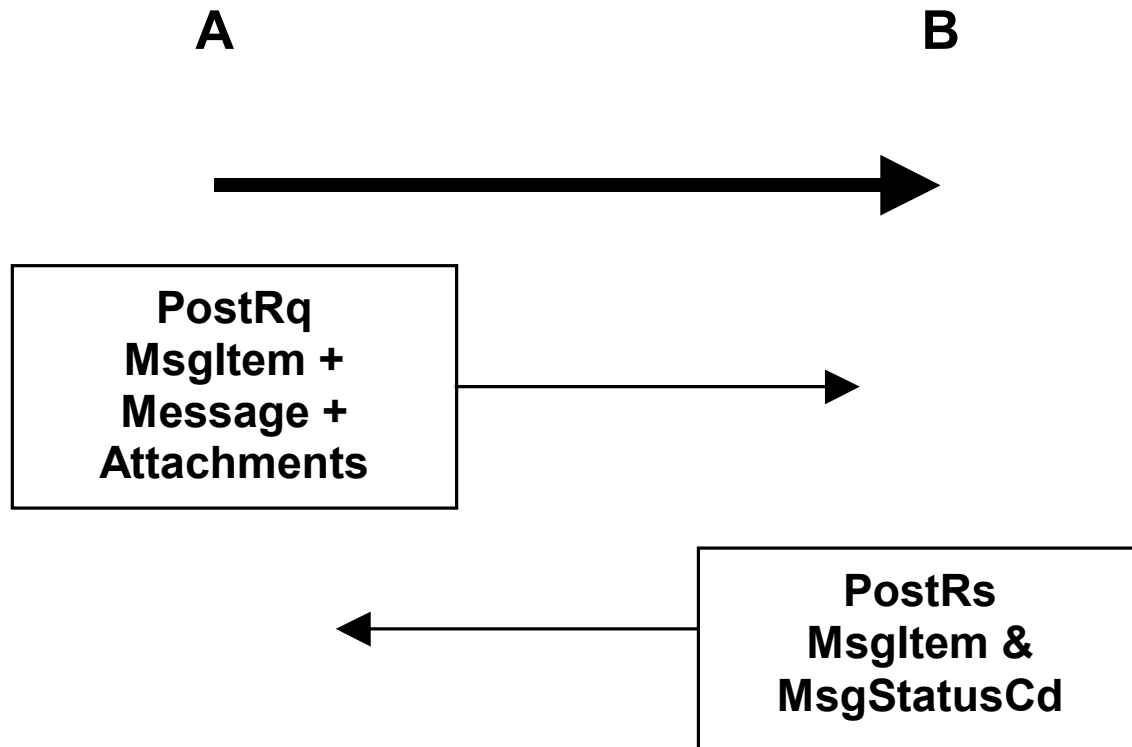
- SOAP 1.1
- MIME
- HTTP 1.1
- SSL/TLS (optional)

## Transport Operation

## Work Folder

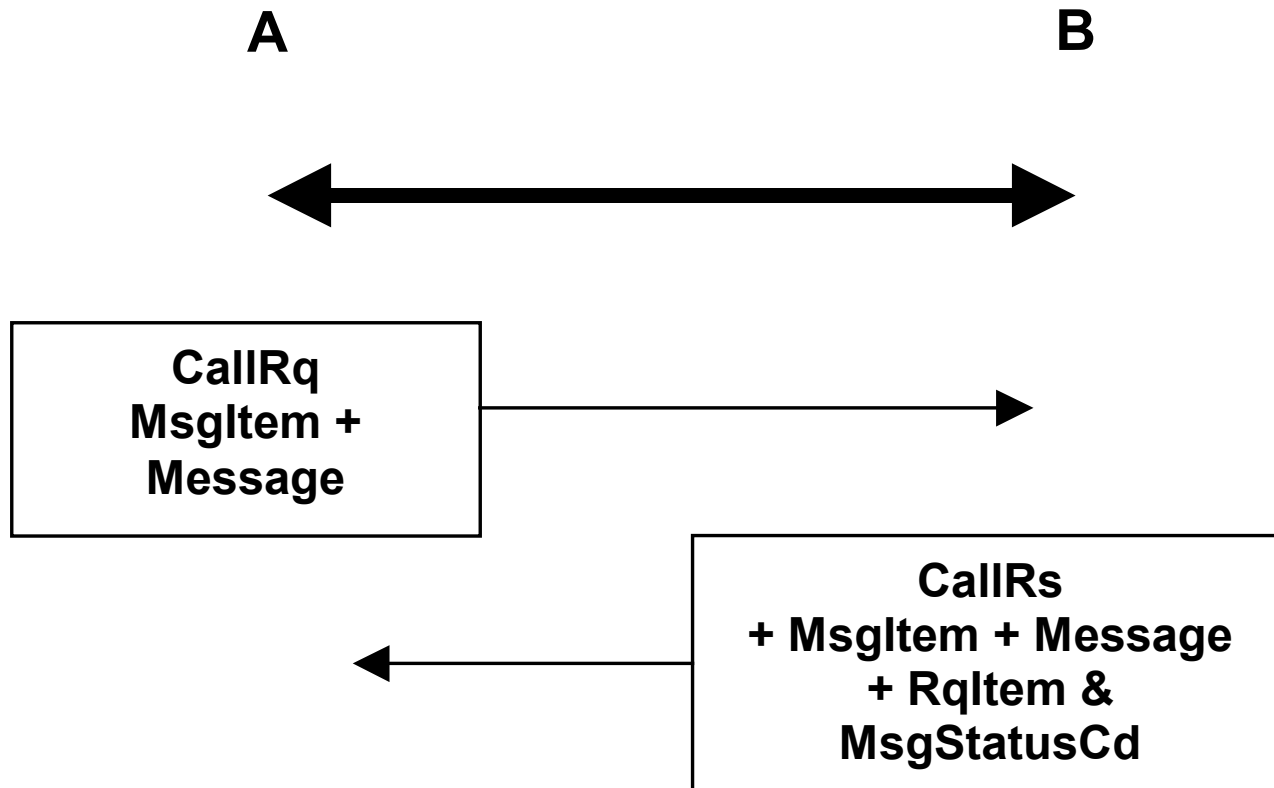
# RLC example

- **One-way ACORD RLC Placing transaction with Attachments**



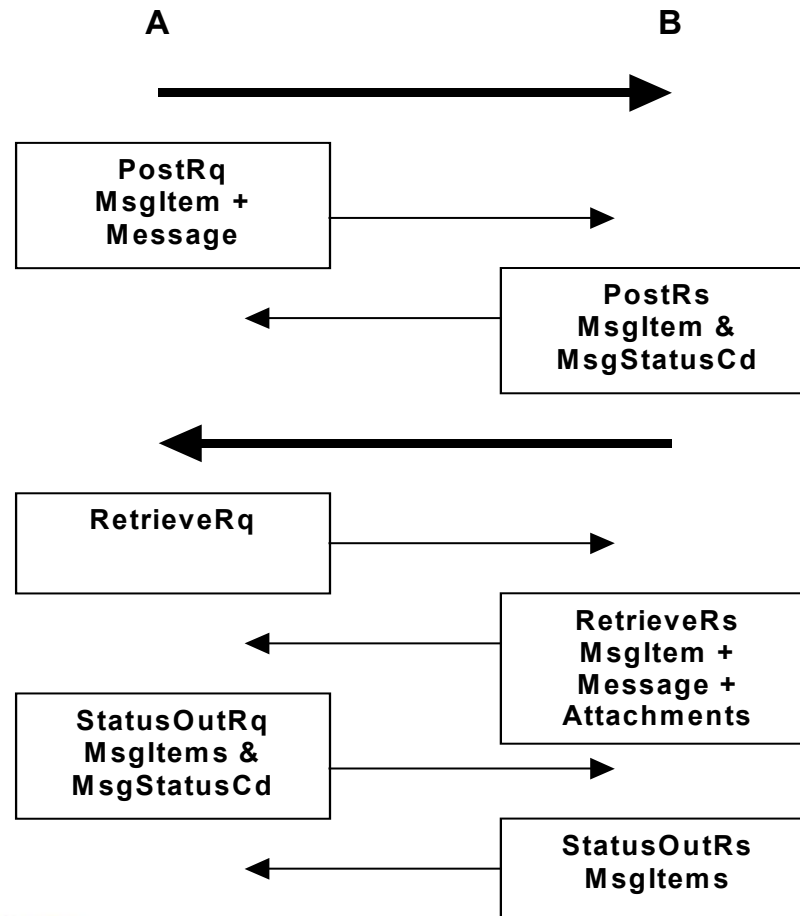
# P&C example

- **ACORD P&C PersAutoPolicyAddRq**  
synchronous transaction



# Life example

- **ACORD Life “204 - Party Inquiry” asynchronous transaction with Attachments**

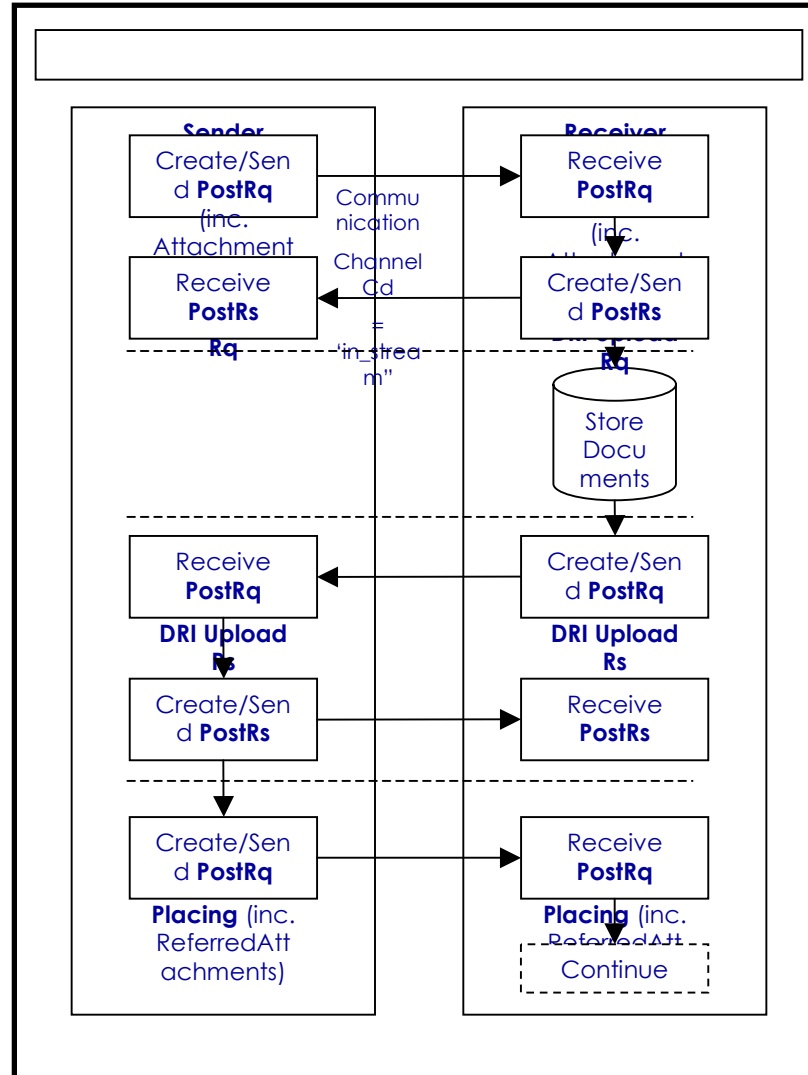


# Document Repository Interface (DRI)

---

- **Complementarity and consistency**

# DRI example



# To get the specification

---

- [http://www.acord.org/Standards/download\\_standards.aspx](http://www.acord.org/Standards/download_standards.aspx)
  - RLC Page
  - ACORD Messaging Service - SOAP Implementation Guide v1.0.1

# Implementations



## Brixlogic

**Composite Web Services Platform:**  
A Unique Web Service Enablement Solution

- ▶ **Fastest Time-to-Market:** Don't code, instead, visually compose from industry-specific **certified** components (ex. ACORD® SOAP Framework, IFX support).
- ▶ **Highest Flexibility:** Unique real-time validation technology uses XML Schemas at design time to prevent/detect errors and perform change impact analysis.



# Implementations

---

## Public Use Cases

### ▶ Banking

- **Who:** **DIEBOLD**, #1 Global Financial Self-Service
- **What:** Integration of ATM and Branch banking software through unique IFX-based Web Services.

### ▶ Insurance

- **Who:** **ACORD**
- **What:** Developed an implementation of the ACORD SOAP Framework that can be used for certification.

# Implementations

## EnabledB2B Product



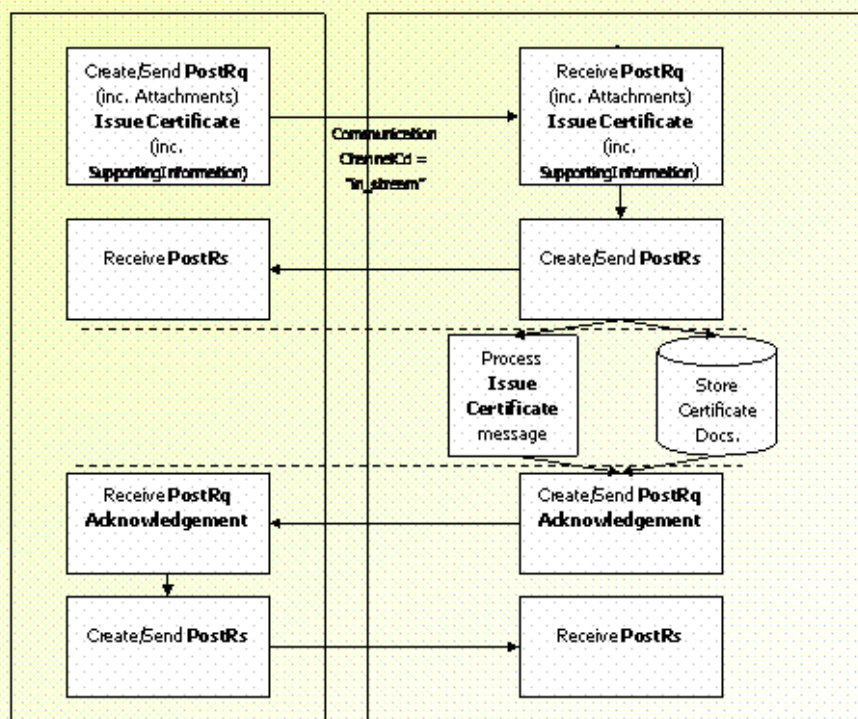
- A 'Black Box' packaged ACORD Messaging Solution
- Certified for ACORD SOAP Framework Call & Post Operations
- Requires minimal integration to internal applications
- Fully configurable to enable compliance with Trading Partner Agreements
- Provides Code translation
- *Greatly reduces the **Time**, **Cost**, and **Risk** of implementing ACORD compliant Messaging Solutions*

# Implementations

## Post Operation Example



### Sample Use Case for Issue Certificate with Documentation



# Implementations

## Softshare XML Business Framework Solution

- Softshare ECS 4.0 and Softshare Delta 4.0 make a rapid development environment for Web Services:
  - Out-of-the-box solution
  - Any-to-any, drag-and-drop mapper
  - Supports SOAP (with attachments), HTTPS, EDI, XML, database access and flat files
  - No custom Java or .NET coding

Softshare DELTA<sup>®</sup>      Softshare ECS<sup>®</sup>

← Softshare

# Implementations

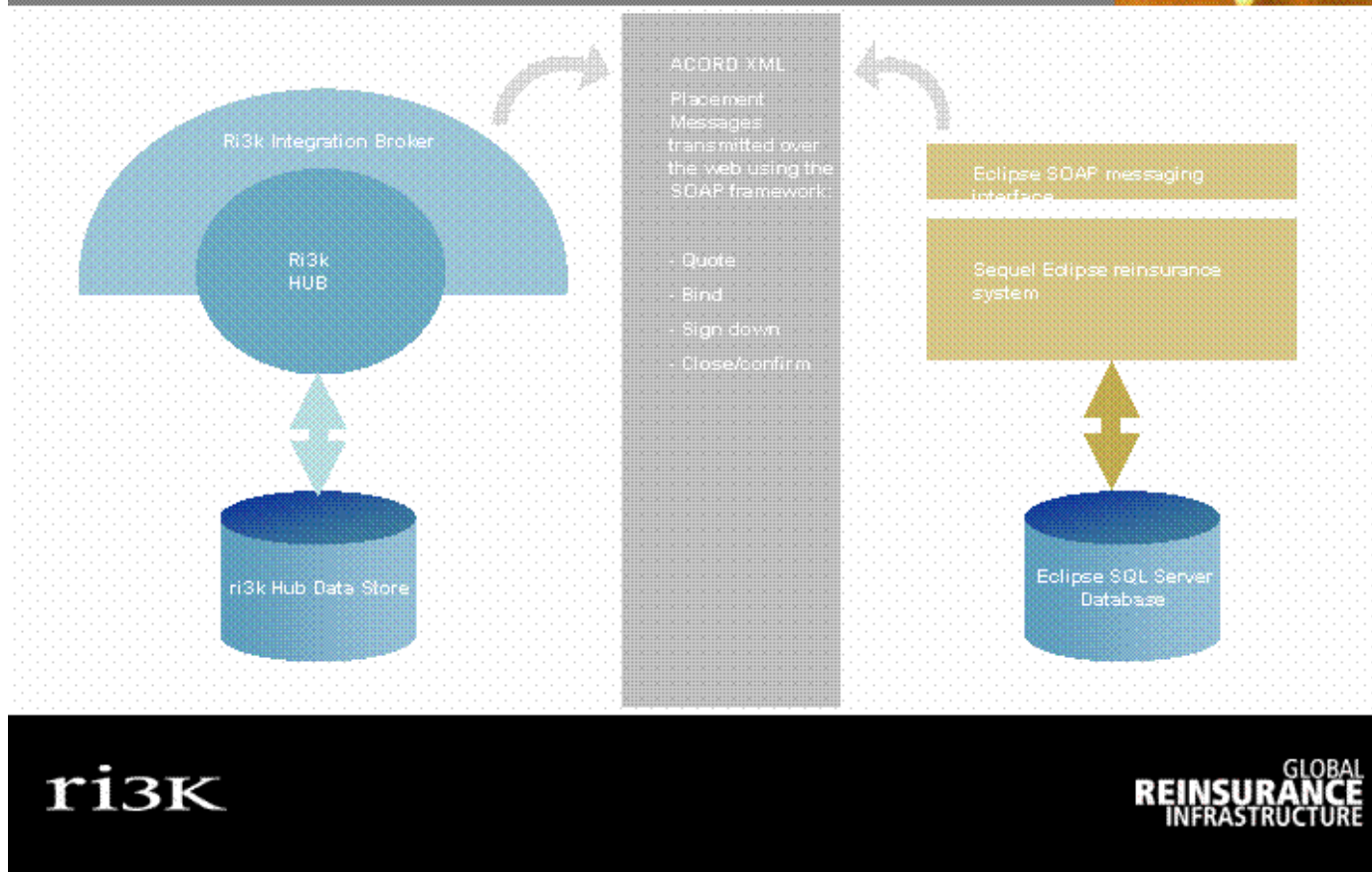
## ACORD SOAP Messaging using Softshare's Products

- Exchanged ACORD business documents over HTTP/S using SOAP with Attachments:
  - Re-insurers (XL Re and SCOR)
  - One broker (Guy Carpenter)
  - Baseline test sites (Softshare and IEC)
- Two of the three ports were implemented synchronously: Inbox and Call using transport service protocol operations: PostRq/PostRs, CallRq/CallRs

↩ Softshare

# Implementations

A real world implementation of the ACORD SOAP framework:  
Sequel Eclipse system – ri3k reinsurance trading platform



ri3K

GLOBAL  
REINSURANCE  
INFRASTRUCTURE

# Implementations



## iVamp: Insurance Value-added Message Processing

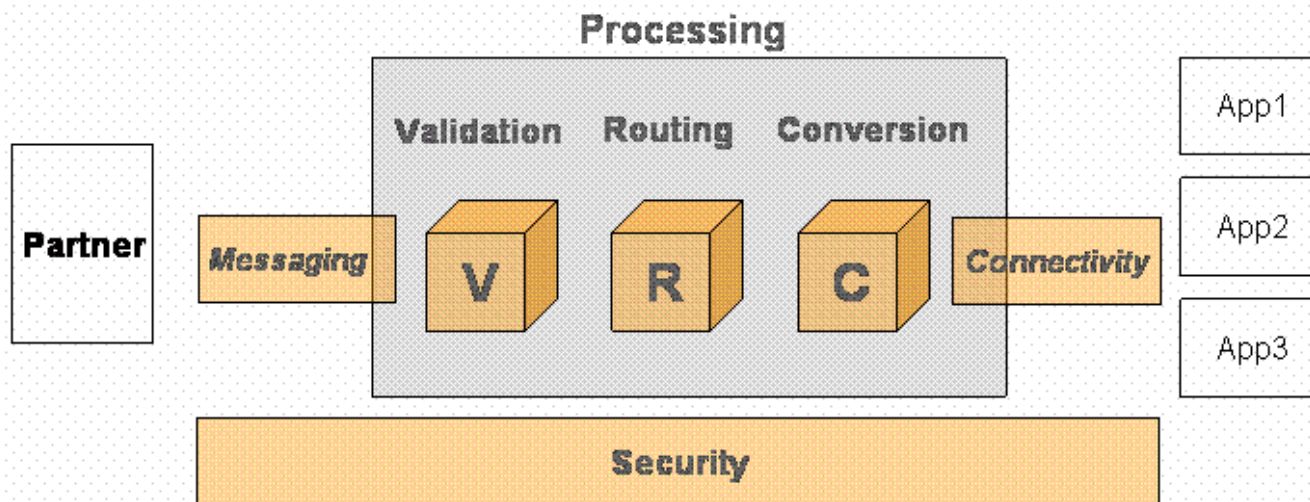
- *A standards based messaging service for reliable and secure exchange of electronic data with business partners over the Internet*
  - Enabling direct collaboration between multiple enterprises without intermediaries
  - Providing automated exchange of structured and unstructured content at no costs per transaction
  - Compliant with and certified by ACORD for the SOAP Messaging Service Specification
  - Flexible conversion services or backend connectivity adapters available as needed
  - Delivered on cost-effective, robust and expandable communication infrastructure

[seite 2]

# Implementations



## The six iVamp feature areas



[seite 3]