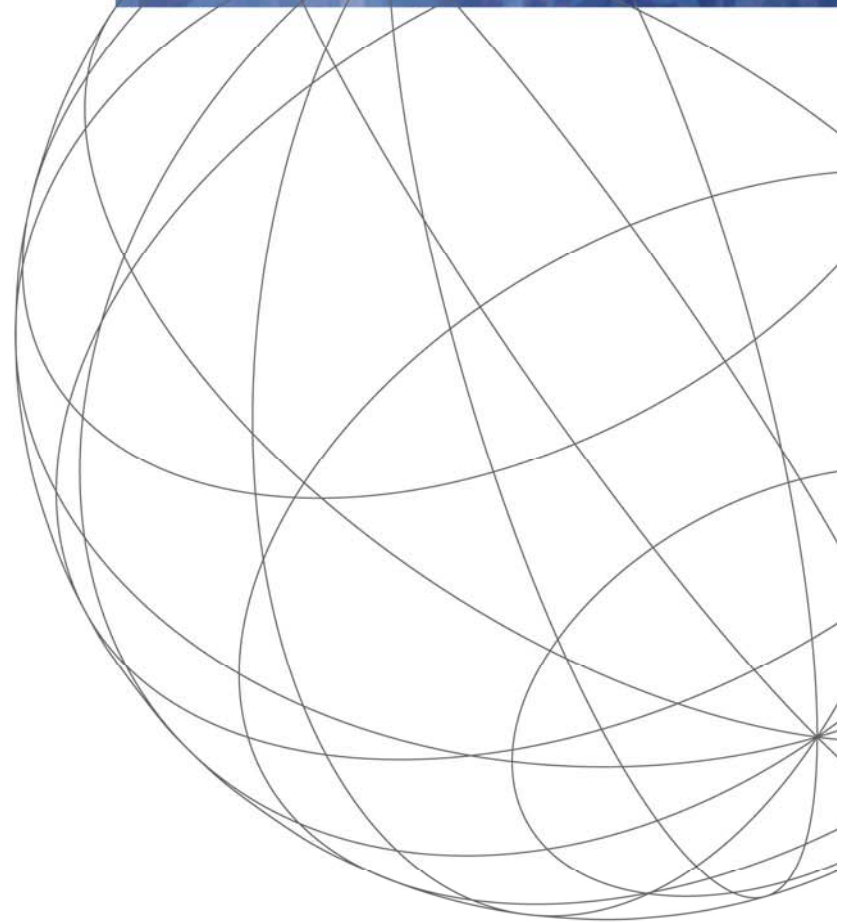




# ACORD Web Service Profile overview

## Insurance Systems Forum – May 2006

Serge Cayron  
Technical Architect



# Background

---

- **Horizontal cross-industry standards is a critical foundation**
  - for interoperability of platforms
  - for deployment of vertical business standards
- **Initial achievements**
  - First step in Web Services standards application : *ACORD SOAP Messaging Service (**AcordMsgSvc**) version 1*, released in 2003.
  - Successfully implemented by a number of members in the Reinsurance community in Europe and the US.
  - Mandated for all London Market RLC and DRI implementations (up to 300 participants)



# Business case

---

- **New generation of cross-industry WS- Standards (around SOAP 1.2), covering Insurance Industry's framework requirements more comprehensively.**
  - Lowers investment in ACORD specific features in the area of frameworks,
  - Ensures widest applicability of framework standards for all applications in Financial services
- **Need to support ACORD Standards strategy, which adopted SOA (Service Oriented Architecture) as a key paradigm.**



## Business Case (cont.)

---

- **Timely development of PCS V2, needing an implementation pilot on a Web Services framework in 2006.**
  - Build business data message architecture on optimal set of Web Services features
- **Desire to work on a specification, commonly adopted by the 3 ACORD constituencies,**
  - In particular align more closely to conventional SOAP message design, with business payload in the SOAP Body.



# Joint working group

---

- **Joint Technical – Web Services Profile**
- **Co-chairs**
  - PCS: Sandi Perillo, The Hartford
  - Life: Jim Brain, Aegon
  - RLC: Andrew Cassels, Lloyd's



# Obstacles toward a joint profile?

---

- **Web Services deal with a lower layer of system interfacing than business messages.**
  - Application of web service standards should have no impact on usage of the actual business messages.
- **PCS V2 wants an optimal Web Services design: is this in conflict with the legacy of the 2 other domains?**
  - No required features identified for the profile would make it incompatible with the current Life or RLC standards.
- **Possibility of duplicating features already covered in the Life or RLC standards**
  - We cannot change the Life and RLC transaction wrappers, but duplicating features is NOT incompatible



# Charter

---

- **Functionality in scope (AcordMsgSvc 1.X inclusive)**
  - Business message wrapper
  - Attachment exchange
  - End-to-end addressing
  - Messaging patterns
    - ▶ Request/Response
    - ▶ One-way
  - Messaging Security
  - Reliable and ordered transport
  - Guidance on Web Services granularity
  - Support processing models including intermediaries
- **Out of scope**
  - Standard management of transaction choreography



## Charter (2)

---

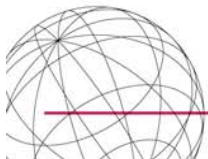
- **Leverage completed Web Service Standards and profiles**
  - W3C: SOAP 1.2, MTOM, WS-Addressing, WSDL
    - ▶ For short term implementations, define a migration path allowing for SOAP 1.1 based development
  - OASIS: WS-Security and WS-ReliableMessaging
  - WSI and OASIS profiles
- **ACORD “profile”**
  - Recommended way of applying WS standards to Insurance industry use cases.
  - Recommended way of composing autonomous WS standards, for simplification and interoperability.



## Charter (3)

---

- **Feed Industry requirement into W3C, OASIS and WSI**
  - avoid ACORD specific whenever possible
  - via appropriate liaison with these standard bodies

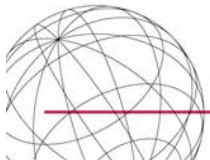


# Work plan

---

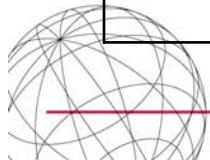
## ■ Two releases:

- AWSP 1.0, for immediate deployment on existing platform versions (SOAP 1.1 generation)
- AWSP 2.0, for deployment with target standard versions (SOAP 1.2 generation)
  - ▶ Not all target standards are ready
  - ▶ Cross-Industry profiling work just starting.



# Work plan – Payload independent components

| Component                  | AWSP 1.0, based on                                                                                                                                                                                                                                                         | AWSP 2.0, based on                                                                                                                                                                                                  |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Basic SOAP + WSDL Profile  | <p>Cross-industry<br/>           SOAP 1.1<br/>           WSDL 1.1<br/>           SOAP with Attachments (SWA)<br/>           WSI-Basic Profile</p> <p><b>ACORD</b><br/>           Service granularity rules<br/>           Message Exchange Pattern options</p> <p>2006</p> | <p>Cross-industry<br/>           SOAP 1.2<br/>           WSDL 2.0<br/>           MTOM</p> <p><b>ACORD</b><br/>           Service granularity rules<br/>           Message Exchange Patterns options</p> <p>2007</p> |
| Addressing Profile         | <p>Cross-industry<br/>           WS-Addressing</p> <p>2006</p>                                                                                                                                                                                                             | <p>Cross-industry<br/>           WS-Addressing</p> <p>2007</p>                                                                                                                                                      |
| Security Profile           | <p>Cross-industry<br/>           WS-Security 1.0<br/>           WSI-Basic Security Profile</p> <p><b>ACORD</b><br/>           ACORD Security Profile</p> <p>2006</p>                                                                                                       | <p>Cross-industry<br/>           WS-Security 1.1<br/>           WSI-Basic Security Profile</p> <p><b>ACORD</b><br/>           ACORD Security Profile</p> <p>2007</p>                                                |
| Reliable Messaging Profile | N/A                                                                                                                                                                                                                                                                        | <p>Cross-industry<br/>           WS-RM</p> <p>2007</p>                                                                                                                                                              |



# Work plan – Payload related components

| Component                | AWSP 1.0 and 2.0 , based on                                                                                            |
|--------------------------|------------------------------------------------------------------------------------------------------------------------|
| SOAP Fault usage         | ACORD<br>Infrastructure error reporting rules<br>ACORD custom SOAP Fault construct<br>2006                             |
| Attachment handling      | ACORD<br>Specification of attachment exchange methods.<br>Attachment manifest construct<br>2006                        |
| Message Groups & batches | ACORD<br>Specification of Message Groups & Batches handling methods.<br>Groups & Batches construct (if needed)<br>2006 |



# AWSP 1.0 (2006)

---

- Basic SOAP (1.1) +WSDL (1.1) profile
- Addressing profile
- (Message groups and batches)
- SOAP Fault usage
- Attachment Exchange
- Security profile



# AWSP 1.0 - Basic SOAP + WSDL profile

---

## ■ Basic Profile:

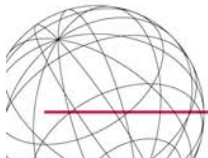
- Core set of specifications that provide the foundation for Web services

## ■ SOAP (Simple Object Access Protocol)

- Serves as the envelope that wraps around the XML payload message
- Together with underlying transport protocol (e.g. HTTP) connects together peered computing systems to enable transactions

## ■ WSDL (Web Service Description Language)

- Describes the service operations offered by an application service provider
- Describes the SOAP message exchanges associated with these operations
- Describes the connecting parameters



# Basic Profile – Hierarchy of specifications

---

## ■ Native specifications

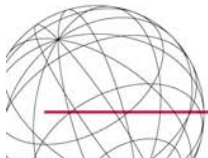
- SOAP 1.1
- HTTP 1.1
- SOAP Message with Attachments (Swa)
- RFC204X: MIME
- WSDL 1.1
- XML 1.0
- XML Schema
- (UDDI version 2)

## ■ WSI Basic and Attachment Profiles

- Generic interoperability
- Using these specifications together in ways that ensure interoperability across platforms, applications and programming languages

## ■ ACORD Basic Profile

- Additional recommendations for Insurance industry use cases



- **Basic Profile**

- ▶ Basic Profile 1.0 and 1.1

- More than 200 interoperability issues resolved in the Basic Profile 1.0; conventions around messaging, description and discovery

- ▶ Simple SOAP Binding Profile 1.0

- Derived from Basic Profile requirements related to serialization of an envelope and its representation in the message

- ▶ Sample Applications and Testing Tools for the Basic Profile

- **Attachments Profile 1.0**

- ▶ Complements the Basic Profile 1.1 to add support for conveying interoperable, SOAP with Attachments (SwA) with SOAP messages

# Reuse of the WSI Basic Profile 1.1

---

- **Basic Profile 1.1**
  - SOAP 1.1
  - WSDL 1.1
  - UDDI version 2
- **Attachment Profile 1.0**
  - SOAP with Attachments (Swa)
- **Scope of reuse**
  - WS-I Basic Profile interoperability rules will be mandatory
    - ▶ Note: UDDI is out of scope
  - ACORD Profile will give guidance on WS-I conformance testing



# ACORD Basic Profile

---

## Additional guidance required for:

- **Service Operation granularity rules**
  - Give guidance on how ACORD business messages and business processes should be related to web service operations.
- **WSDL message exchange patterns options**
  - Give guidance on how WSDL message exchange pattern options map to ACORD message flows.



# Service granularity rules

---

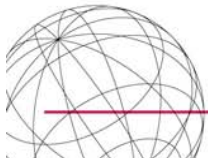
- ACORD has not yet standardized processes solidly and AWSP should not dictate how to map services to existing business messages
  - but this might be a topic for implementation guides in each domain.
- AWSP responsibility is to guide on WSDL design and to develop WSDL samples
  - to reflect transaction one-way and request/response exchange patterns
  - to show WS protocol features (WS-Security, etc.)
  - in each domain + DRI
- Give criteria to define 'homogeneous' services to leverage WS features optimally, e.g for grouped messages.



# WSDL message exchange patterns in AWSP 1.0

---

- **Analysis of message flows**
  - AWSP should start by mapping one-way and Request/Response (synchronous and asynchronous) flows to WSDL Message Exchange Patterns.
- **Asynchronous request-response: two options considered**
  - Use WS-Addressing to correlate a response or a SOAP Fault with the request message
  - Use a robust two-way SOAP exchange for each one-way business exchange.
- **Distinguish symmetric and asymmetric configurations**
  - Symmetric HTTP client/server set up
    - ▶ pushes messages only
  - Asymmetric set-up (fixed client and server)
    - ▶ pushes and pulls messages.
    - ▶ Not sure to find a solution for asymmetric set-up with native WSDL 1.1



# SOAP Basic example (SOAP 1.1)

---

```
<SOAP:Envelope
  xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope"
  <SOAP:Body>
    <TXLife xmlns="http://ACORD.org/Standards/Life/2">
      <UserAuthRequest/>
      <TXLifeRequest>
        <TransRefGUID>01ff00c1</TransRefGUID>
        <TransType tc="204"/>
        <OLife Version="2.8.00"/>
      </TXLifeRequest>
    </TXLife>
  </SOAP:Body>
</SOAP:Envelope>
```



# AWSP 1.0 - Addressing Profile

---

- **WS-Addressing provides transport-neutral mechanisms to address Web services and messages**
  - support message transmission through processing nodes such as endpoint managers, firewalls, and gateways
- **Addressing Profile likely to be optional and minimal in AWSP 1.0**



# SOAP example with Addressing

---

```
<SOAP:Envelope xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <SOAP:Header>
    <wsa:MessageID>uuid:01ff00c1</wsa:MessageID>
    <wsa:To>http://receiver.example/Admin</wsa:To>
    <wsa:From>
      <wsa:Address>http://business456.example/client1</wsa:Address>
    </wsa:From>
    <wsa:Action>http://receiver.example/Admin#tc-204</wsa:Action>
  </SOAP:Header>
  <SOAP:Body>
    <TXLife xmlns="http://ACORD.org/Standards/Life/2">
      <UserAuthRequest/>
      <TXLifeRequest>
        <TransRefGUID>01ff00c1</TransRefGUID>
        <TransType tc="204"/>
        <OLife Version="2.8.00"/>
      </TXLifeRequest>
    </TXLife>
  </SOAP:Body>
</SOAP:Envelope>
```



# AWSP – Message grouping and batches

---

- **Two types of grouping identified**
  - Batches of unrelated messages, e.g. generated as part of periodic process.
  - Logical group: messages needing to be processed together to form a transaction.
- **Implementation options:**
  - Send in a single SOAP envelope (requiring a payload wrapper)
  - Send as an ordered sequence of SOAP messages (with forward view to WS-Reliable Messaging)



# AWSP – Message grouping and batches (cont.)

---

## AWSP current recommendation

- **On batch envelope**
  - Leave the domain standard decide whether a batch envelope is necessary for logical processing reasons or because it fits better the process in place.
  - Specification of batch envelope is in the domain of business message architecture
- **On sequencing**
  - Give guidance on message sequencing with forward view to WS-Reliable Messaging
- **For the future, investigate WS-Transaction for logical groups**
  - Transaction coordination would be handled by infrastructure standards and segregated from pure business standards



# AWSP - SOAP Fault usage

---

## ■ Profiling required

- Define the limits of error reporting by SOAP Faults (up to application infrastructure errors)
- Define ACORD specific fault detail construct



# SOAP Fault example (SOAP 1.1)

---

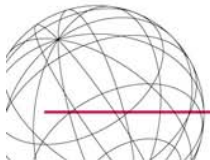
```
<SOAP:Envelope
  xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP:Body>
    <SOAP:Fault>
      <faultcode>Client</faultcode>
      <detail>
        <ac:ErrorReport xmlns:ac="URI">
          <ac:ErrorCode>security</ac:ErrorCode>
          <ac:ErrorDescription>revoked
cetificate</ac:ErrorDescription>
        </ac:ErrorReport>
      </detail>
    </SOAP:Fault>
  </SOAP:Body>
</SOAP:Envelope>
```



# AWSP – Attachment Exchange

---

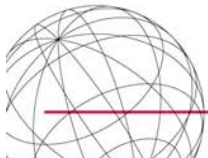
- **Current methods of transport (5 today)**
  - in stream: in the same physical message as the business message(s)
    - ▶ XML embedded encoded binary
    - ▶ MIME parts
  - URL retrieval: documents retrieved
  - Parallel unsolicited transfers: e-mail, ftp etc.
  - DRI transfer: upload or download
  - Life <FormInstanceRequest>
- **Goals**
  - Unification of in stream transport methods (target is MTOM)
  - Rationalization of alternate transport methods
  - Unification of attachment descriptors
    - ▶ Segregation of document attributes from business level metadata
  - Unification of attachment security protection



# Attachment Exchange – AWSP scope

---

- **Inband transport and encoding**
  - Specify how an Attachment File is encoded and transported with a SOAP message (e.g. using SOAP+MIME and/or MTOM)
- **Generic scenarios for attachment exchange**
  - Specify the scenarios of attachment exchange in relation to messages (in-band, out-of-band pull, push etc.)
- **Common data set for attachments**
  - Specify the common data set to support these scenarios
    - ▶ XML components that would be used in messages to enable attachment exchange.
- **Abstract interface approach to enable attachment exchange**
  - with consideration of exchange scenarios,
  - but without consideration of implementation details of the attachment exchange process.



# Attachment Exchange - vision

---

- Encourage decoupling of service requirements, to enable the infrastructure consuming more and more functionality not part of the core business.



# MTOM and forward view to it

---

- **MTOM only available with SOAP 1.2**
  - In practice, only available in very new tools and linked to SOAP 1.2 features
- **AWSP 1.0: support the Swa option and/or the embedded file option.**
  - This is to allow Trading Partners to agree on a trade-off between forward compatibility and immediate performance.
  - Also allows for early MTOM usage between homogeneous platforms.



# AWSP 1.0 – Security Profile

---

- SOAP messaging security, transport and other security considerations
- The ACORD Security Profiles v 1.0 can be reused as is for the initial (SSL/TLS) and basic (Signature of Body + SSL/TLS) levels.



# Security Profile – Hierarchy of specifications

---

## ■ Native specifications

- XML-Signature Syntax and Processing
- XML Encryption Syntax and Processing
- The TLS Protocol Version 1.0
- HTTP over TLS
- RFC2459: Internet X.509 Public Key Infrastructure
- Web Services Security 1.0
- Web Services Security Profiles (X509, Username Token)
- Web Services Security: SOAP Messages with Attachments (SwA) Profile 1.0

## ■ WSI Basic Security Profile

- Generic interoperability
- Using these specifications together in ways that ensure interoperability across platforms, applications and programming languages

## ■ ACORD Security Profile

- Additional recommendations for Insurance industry use cases



- **Basic Security Profile**

- ▶ Security Scenarios (Working Group Draft)
  - Document security risks in interoperable Web services, along with potential countermeasures
- ▶ Basic Security Profile 1.0 (Working Group Draft)
  - Addresses transport security, SOAP messaging security and other security considerations for WS-I Profiles
  - Profiles OASIS' Web Services Security specification
  - Final version expected in early, 2005

# ACORD Security Profile – current AcordMsgSvc version

| Security Technique per Profile level | Initial | Basic  | Medium | Maximal |
|--------------------------------------|---------|--------|--------|---------|
| SSL/TLS Server authentication        |         |        |        | Option  |
| SSL/TLS Integrity                    |         |        |        | Option  |
| SSL/TLS Encryption                   |         |        |        | Option  |
| TLS Client authentication            | Option  |        |        |         |
| WSS User name token                  | Option  | Option | Option | Option  |
| WSS signature of SOAP Envelope       |         |        |        |         |
| WSS signature of all parts (Swa)     |         |        |        |         |
| ACORD Referred Message Signature     |         | Option |        |         |
| ACORD File Signature                 |         |        | Option | Option  |
| WSS encryption of SOAP Envelope      |         |        |        |         |
| WSS encryption of all parts (Swa)    |         |        |        |         |



# Web Services Profile – benefits expected

---

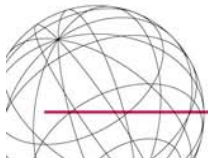
- **Single design for XML payload wrapping and transport**
- **Single design for attachment exchange**
  - Ultimately catered for by MTOM, which unifies the binary attachment methods.
- **Messaging security**
  - Expect off-the-shelf robust security solution
- **Messaging reliability**
  - Expect off-the-shelf robust solution supporting reliable one-way and request response exchanges
- **Guidance for Web Service interface design**
  - Granularity of web service exposed
  - Use of Service port types
  - Single design for message addressing and routing



# Working group details

---

- **Name: Joint Technical – ACORD Web Service Profile**
- **Participants:**
  - PCS: 24
  - Life: 24
  - RLC: 13
  - All : 7
- **If you are planning WS implementation, join the group or get access through Teams**

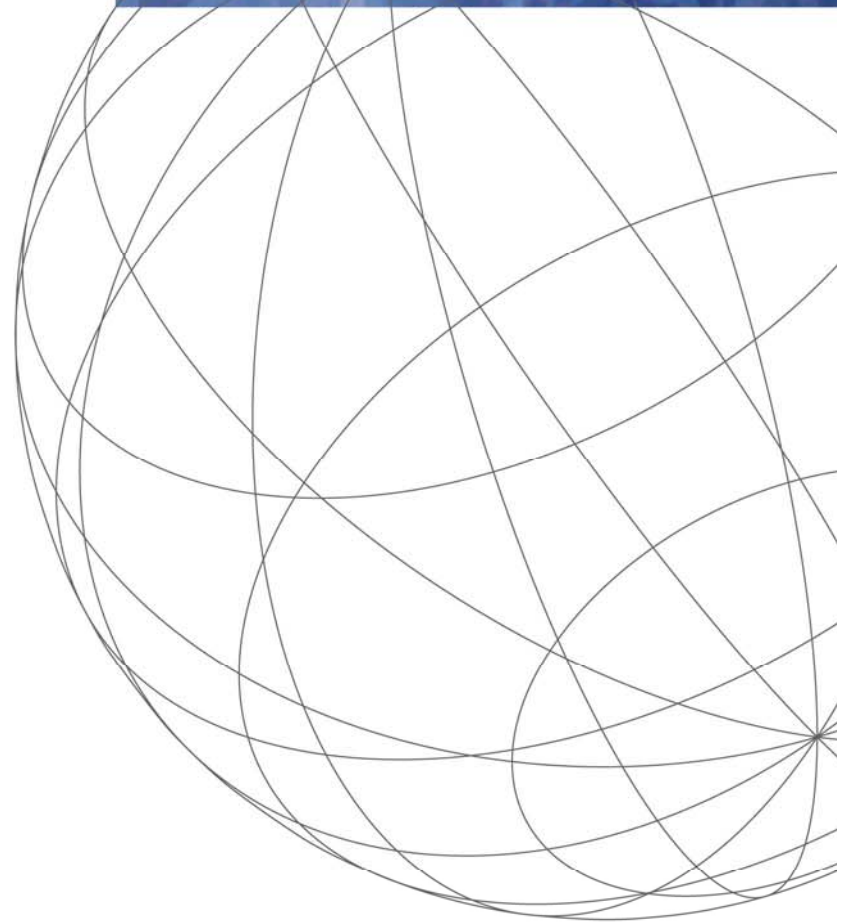




# Introduction to Document Repository Interface Standard

## Insurance Systems Forum – May 2006

Serge Cayron  
Technical Architect



# Content

---

- Business requirements
- Standard specification
- Implementation Guide



---

# Business requirements



# Business pressures and requirements

---

## ■ Business drivers

- Cost of handling paper
- Lack of control
- Lack of availability
- Delays in distribution of paper serially

## ■ Electronic documents

- Support decision making:
  - ▶ Placing/Submissions, Claims, Closing/Accounting
- Easier to share
- Reduce claims advice cycles
- Improve tracking and control



# Business responses

---

- Major investments in Document Management Systems
  - All parties worldwide
- Recognition of multiplicity of repositories
  - And different ways of working
- Need for interoperation = standards



# Which standard is needed?

---

- **Current PCS, RLC and Life standards support attached document referencing in business messages.**
  - Transport framework is assumed to provide a mechanism to exchange attachments with messages, e.g. in MIME packages.
- **Need to allow document exchanges independently of business message exchanges**
  - Direct exchanges between document repositories, with associated business metadata.
  - Parallel and coordinated transfer of business messages and associated documents.
  - With the same level of security (integrity, authentication, privacy) as for business messages.



# Document Repository Interface

---

## Purpose of Standard

- Autonomous, automatic and secure interchange of free-format documents between repositories
- Coupling documents with business data and transactions
- Supporting various interoperation modes



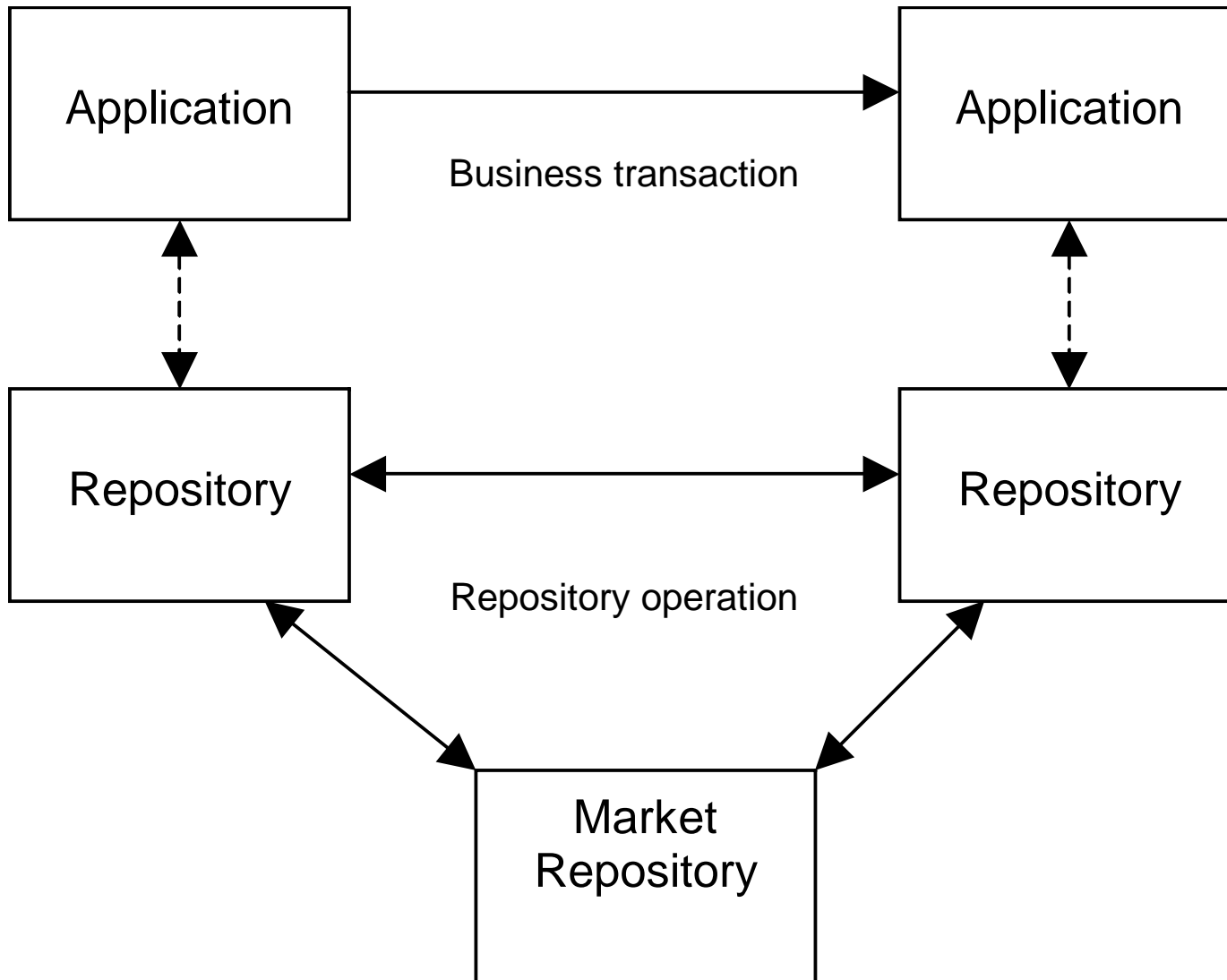
## Implementations (RLC-current and ongoing)

---

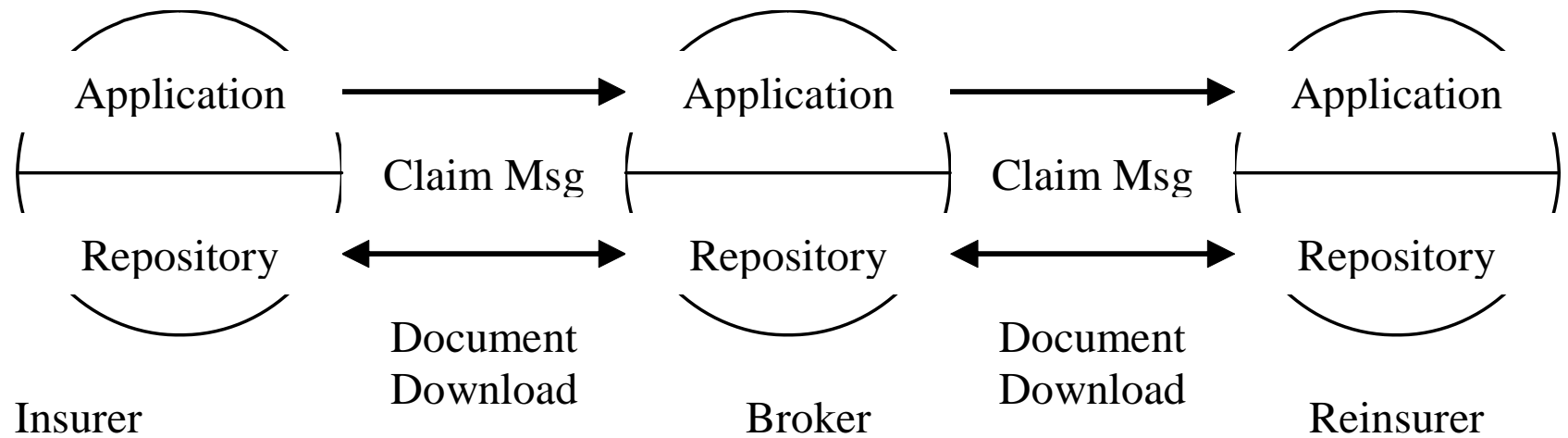
- 5 Brokers
- 6 Carriers
- 3 Solution Providers
- In London, US and Europe



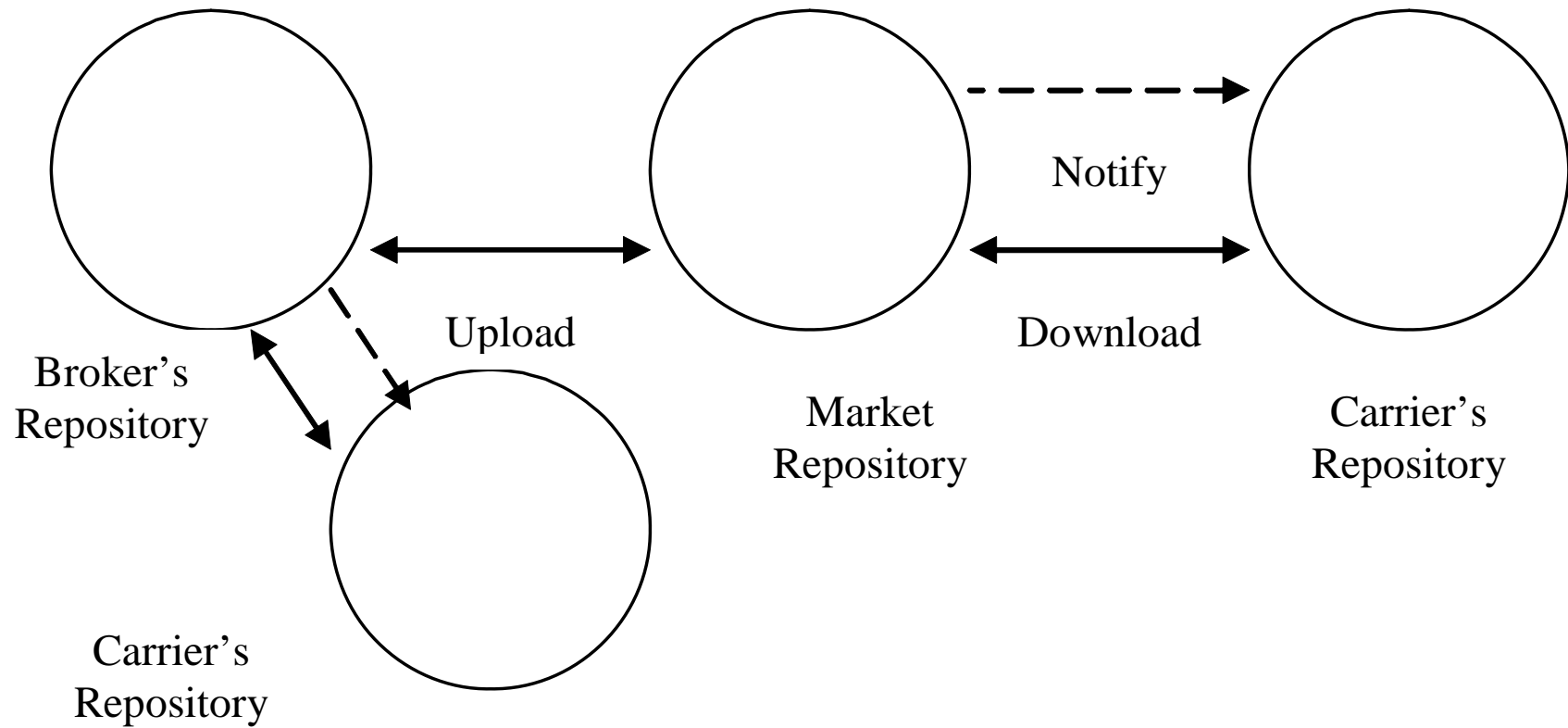
# Interoperating systems



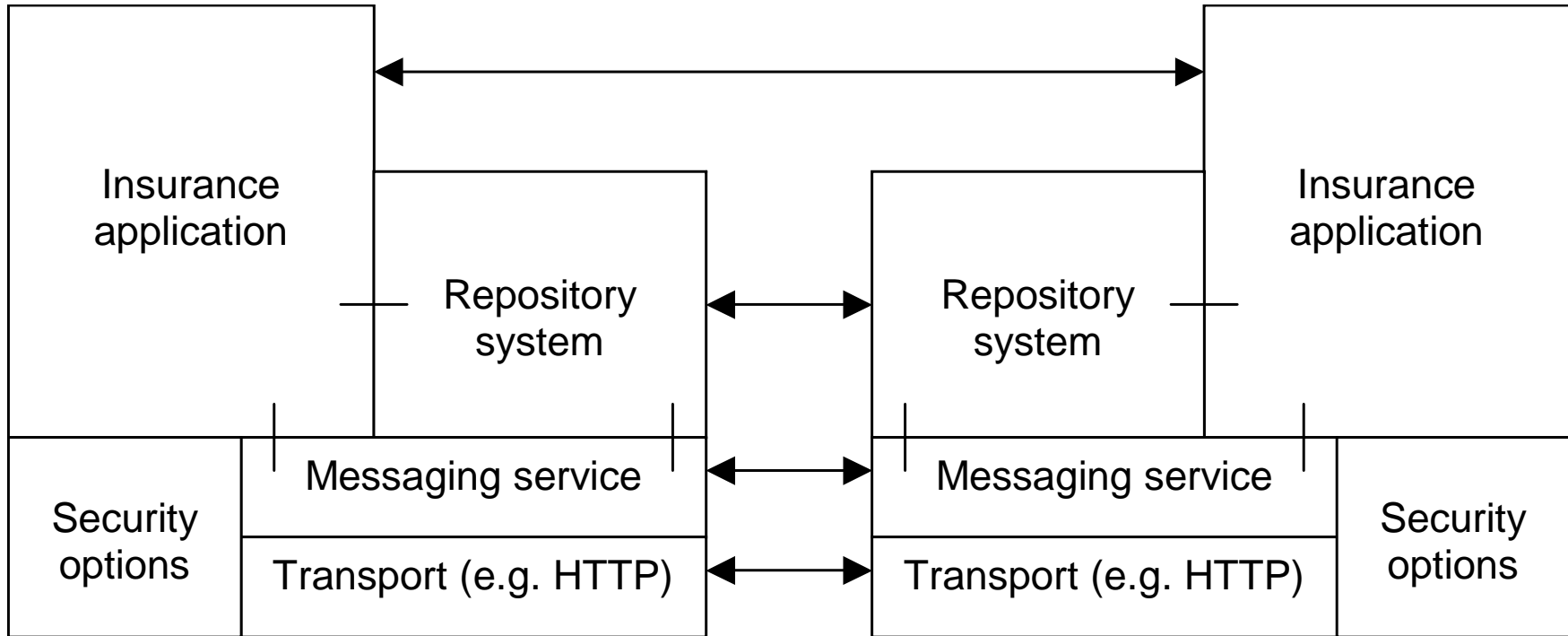
# Use case : US Reinsurance Claims scenario



# Use case : London Market Claims scenario



# Architecture and software components



---

# Standard specification



# DRI standard specification

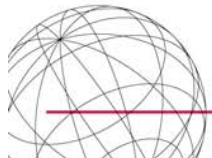
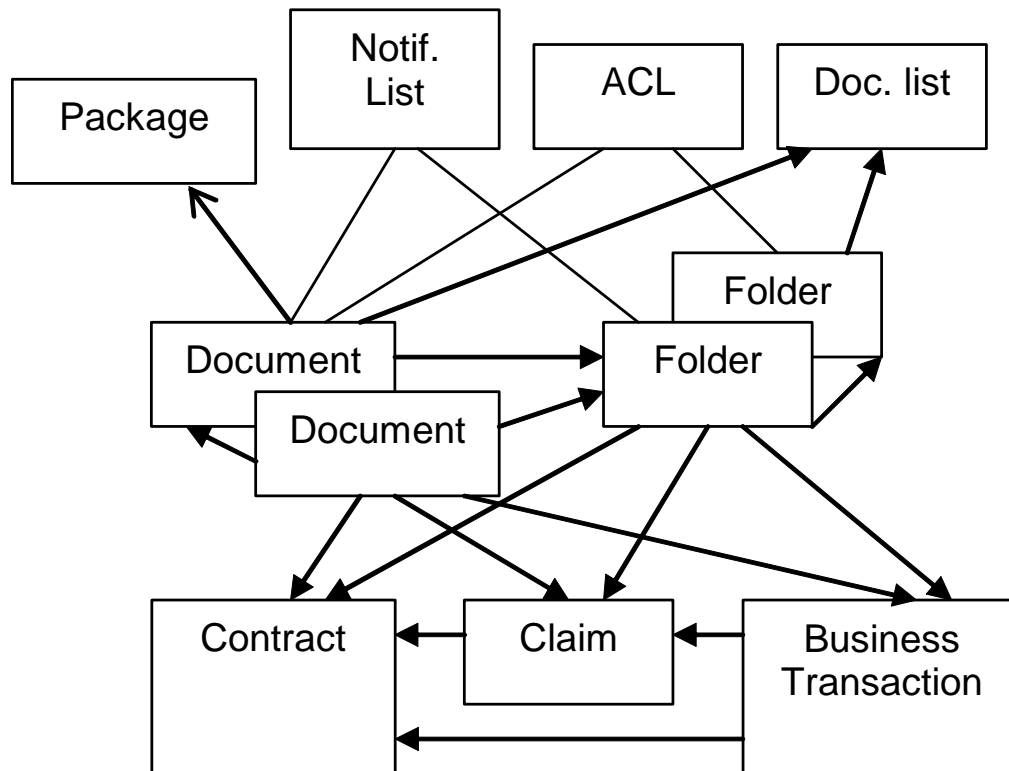
---

- Data model
- Message architecture
- Modes of interoperation – repository operations
- Data requirements per repository operation



# Data model

Party Roles (Originator, Owner, Operator)



# DRI message architecture

---

- **Header**
  - Message id, date, sender, receiver
  - Response info
  - Cross-reference to application (and transaction)
- **Document lists**
  - Sets of document descriptors grouped by Repository Service operator
  - Describing documents and/or folders
  - Each document or folder has:
    - ▶ Identifier and other attributes
    - ▶ Identity of owner (and originator).
    - ▶ Business metadata.
    - ▶ Optional Access Control and Notification lists.
- **Search criteria**
  - Data to control scope of search (max number, date range etc.)



# Document and Folder information

---

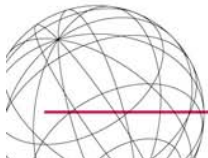
- **Alternate identification options:**
  - UUID
  - Owner assigned reference
  - Owner assigned reference and version (documents only)
- **Folders can be organized in hierarchies and documents can be located in a Folder.**
- **Documents can be further related to one another with a selectable type of relationship (e.g. Annotation, Version, Component)**
- **Other attributes of documents or folders:**
  - Document type code
  - Version date & time
  - URL
  - File name
  - File format (MIME content type)
  - File size



# Business metadata

---

- Specified for each Document or Folder by reusing XML tags from business standard.
- Business vocabulary acquired by composing relevant schema slices of RLC, PCS or Life with the DRI schema.
- E.g. RLC reuses tags for:
  - Party roles (Cedent, Broker, Reinsurer etc.)
  - Policies and Contracts (Contract, ContractSection, OriginalPolicy)
  - Claims (Claim, ClaimEntry)
  - Transaction identifiers



# Request and response messages

---

- Each repository operation is composed of a pair of request and response messages
- All-or-nothing approach regarding to operation request processing (Success/Reject)
- Small selection of data elements from a request message are echoed back in a response message



# Message templates

---

## ■ Show:

- Request: RepositoryOperationRq-template\_v1-0-1.xml
- Response: RepositoryOperationRs-template\_v1-0-1.xml



# Modes of interoperation

---

## ■ Push documents

- Upload to repository through *Upload* operation
  - ▶ Alternately: in stream with business message

## ■ Pull documents – notify and download

- Notification to repository through *Notification* operation
  - ▶ Alternately: notification to application through business message
  - ▶ Alternately: notification to end user through *e-mail or ad-hoc*
- Download to repository through *Download* operation
  - ▶ Alternately: download to end user through web browser



# Modes of interoperation (cont.)

---

## ■ Search and Pull

- Automated search through *Search* operation
  - ▶ Alternately: search by end user through web browser
- Download to repository through *Download* operation
  - ▶ Alternately: download to end user through web browser

## ■ Interoperation with Repository Service

- *Create Folder* operation
- *Download Folder* operation
- *Change Attributes* operation
- Exchange of Access Control Lists and Notification Lists



## Data requirements per repository operation

---

- A Data Requirement Matrix sets out the definitions and usages for all elements of *RepositoryOperationRx* messages.
  - Usages are given in the context of each specific operation code, request and response



---

# Implementation Guide



# DRI implementation guide

---

- Underlying communication framework
- Message flows - Combined Business Message & DRI Message
- Message flows - Standalone DRI Message



# Underlying communication framework

---

- DRI can operate with any communication framework, provided that it supports:
  - exchange of payload messages and documents in a single communication envelope
  - asynchronous and synchronous processing of payload messages
- DRI currently has a binding defined for the ACORD Messaging Service.
- AWSP protocol binding will be specified when needed.



# Message flows - Combined Business Message & DRI Message

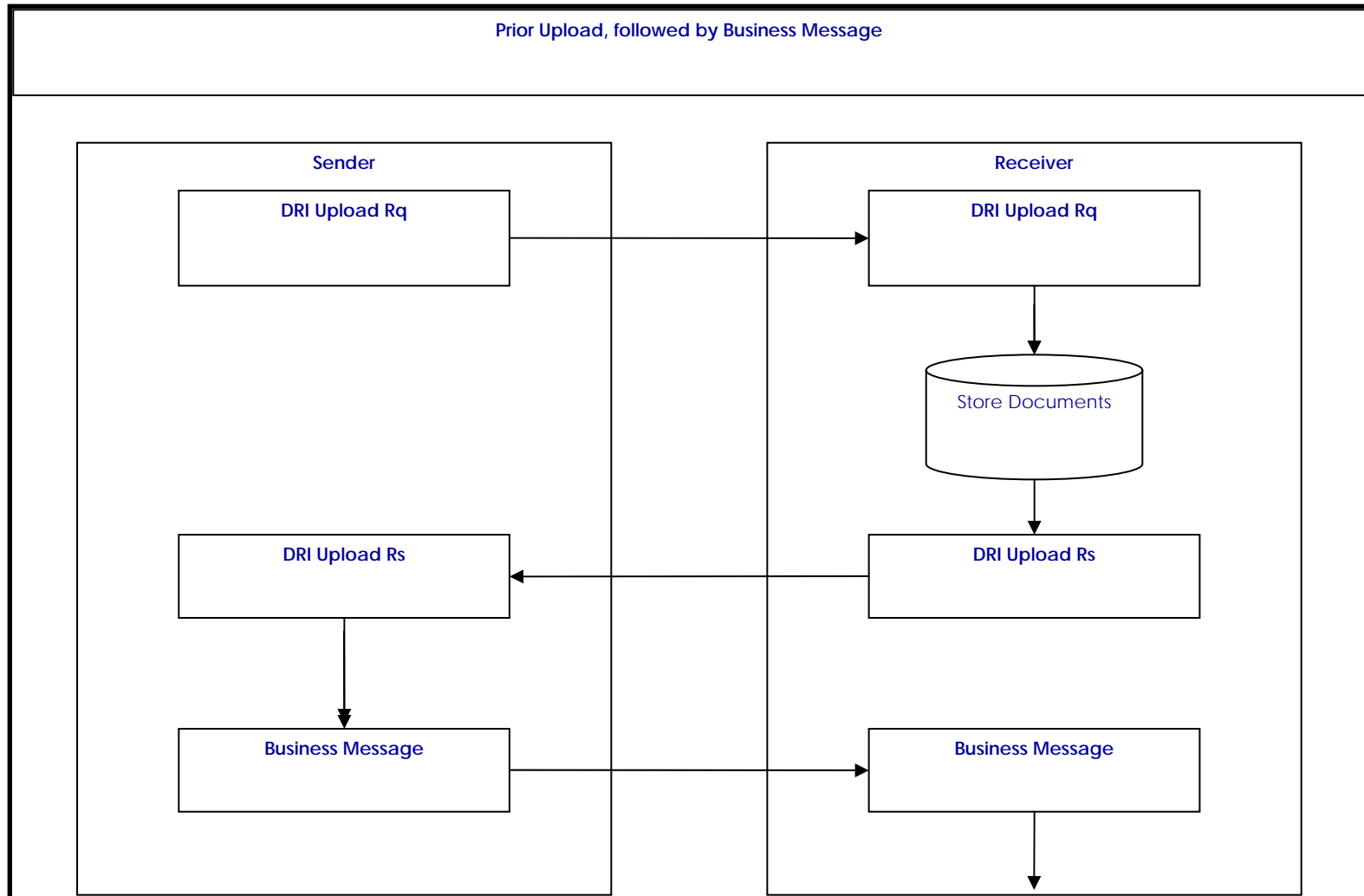
---

## Examples:

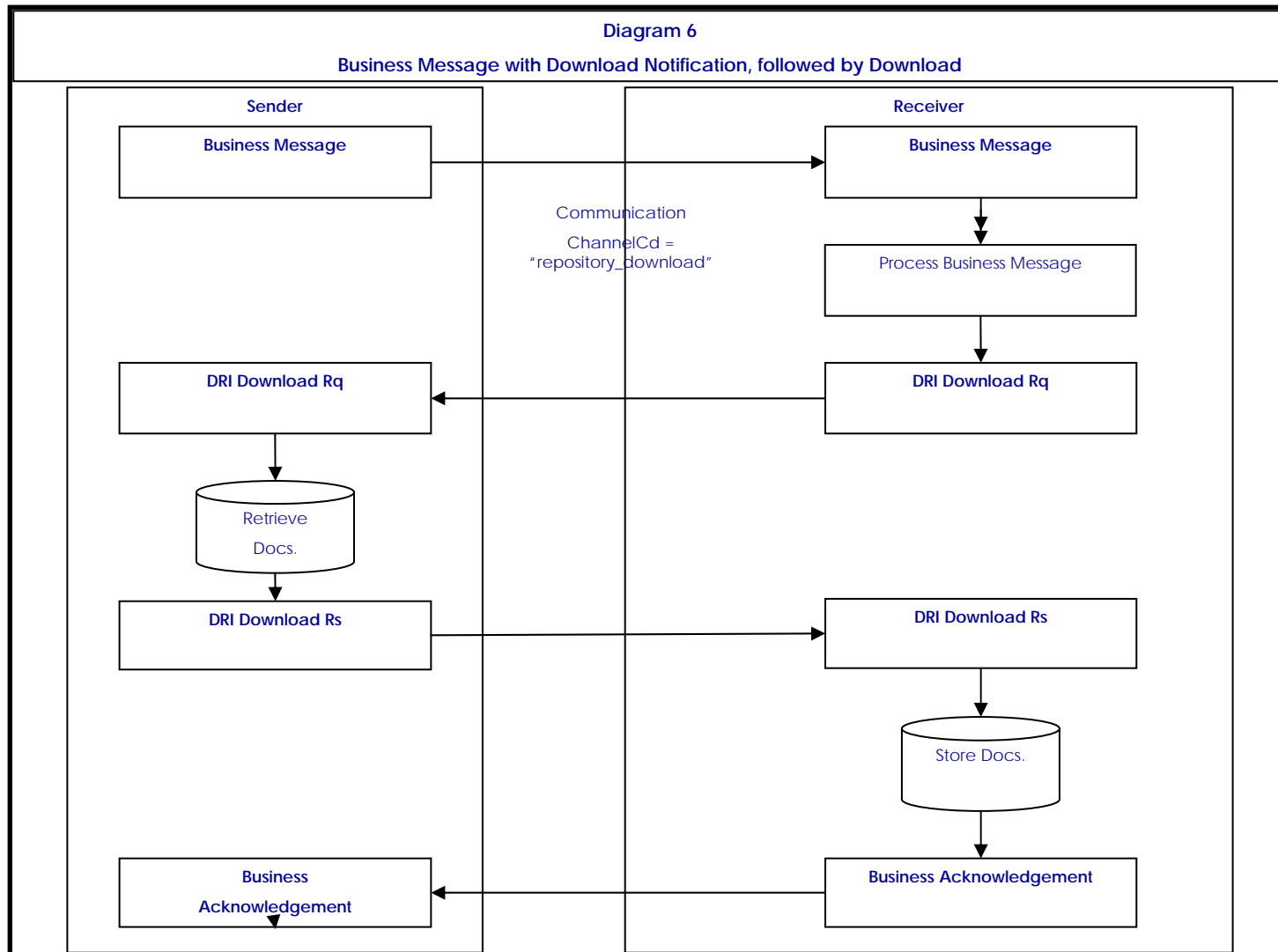
- Prior Upload, followed by Business Message
- Business Message with Download Notification, followed by Download



# Prior Upload, followed by Business Message



# Business Message with Download Notification, followed by Download



# Message flows – Stand alone DRI Message

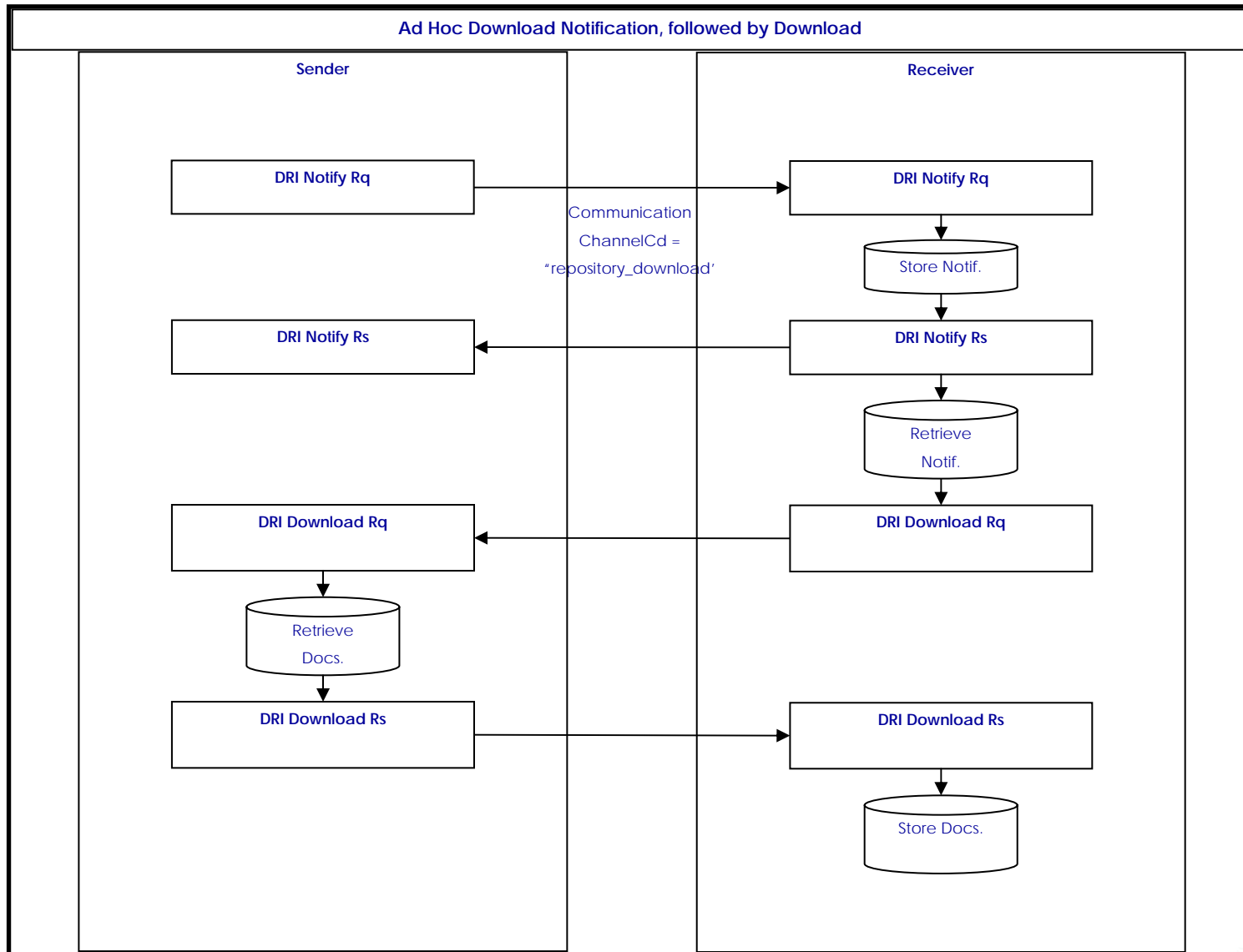
---

## Example

- Ad Hoc Download Notification, followed by Download



# Ad Hoc Download Notification, followed by Download



# DRI Standard documentation

---

- **DRI Reference package (public)**
  - Reference Guide
  - Data Requirement Matrix
  - XML Schema
  - XML Message Templates
  
- **DRI Implementation package (members' benefit)**
  - All of the above
  - Implementation Guide
  - Use cases



# To get the specification

---

- [http://www.acord.org/Standards/download\\_standards.aspx](http://www.acord.org/Standards/download_standards.aspx)
  - Public RLC or PCS sections : [Document Repository Interface \(DRI\) Reference Guide - V 1.2.0](#)
  - Members only RLC, PCS and Life sections: [Document Repository Interface \(DRI\) Implementation/Reference Guide - V 1.2.0](#)

