

Agenda

- **Introduction to ACORD**

 - **Introduction**

 - **What is ACORD**

 - ACORD the organization
 - ACORD the data model

 - **Why use a Standard?**

 - Pros
 - Business
 - Technical
 - Risks

 - **ACORD Data Model**

 - ACORD Life Standards
 - Basic XML Concepts
 - Transaction Architecture (TXLife)
 - Life Object Model (OLifE)

 - **ACORD Strategy**

 - **Final Thoughts**

What is ACORD?

ACORD is the Association for Cooperative Operations Research and Development

From a business perspective, ACORD is...

A way to describe insurance business data entities

A way to describe insurance operations (“transactions”)

An insurance industry standard used as a common business language by thousands of carriers and agencies

From an architectural perspective, ACORD is...

A data model that specifies insurance business data entities

A transaction model that specifies insurance business operations using the data model as input/output parameters

Fully implemented using XML and XML Schema constructs

A data model that provides flexibility for Conesco specific data requirements with an independent extension schema

A stable product - Backwards compatibility is the central design rule for the model

ACORD Standards: An Evolution

Non- Life History

- 1980's – EDI developed
- 1999 – XML developed
- 2004 – New Version of XML developed
- 2008 – New Communities in Australia & South Africa

Life History

- 1996– Acquired standard from Microsoft (OLE Technology)
- 1998 – XML developed
- 2000 – Version 2 developed
- 2001 – New Community in South Africa
- 2011 – New Community in Australia

GRLC History

- 1980's – EDI Started
- 1991 – Created Joint Venture
- 2001 - ACORD
- 2002 – XML developed
- 2008 – New Rüşchlikon Community
- 2010 – New North American

1970's 1980's 1990's 1996 1998 2001 2010 +

Non-life
Forms

AL3 (ACORD's
non-life EDI
standard)

Non-life
Implementation

Life
Standards

XML for
Life & Non-
life
Life
Reinsurance

Non-life
Reinsurance
EDI & XML
Life Forms
Fillable
Forms

Standards
Framework
ACORD
Messaging
Library
eForms

Click on this video link to Learn about
ACORD

<http://link.brightcove.com/services/player/bcpid73529169001?bclid=65696192001&bctid=57767300001>



Why Use a Standard like ACORD?

- **Decrease Costs**

- Reduce time/cost to implement partner interfaces
- Reduce data transaction/translation errors
- Reduce human intervention in routine biz processing

- **Increase Sales/Revenue**

- Make it easier to do business with you
- Decrease time to issue = fewer “not taken”

- **Improve Service**

- Develop “single-view” of disparate systems
- Increase richness of shared information
- Reduce cycle time between upgrades in data share

Pros - Business

- ACORD is an insurance industry standard
 - The ACORD data model has been in use for close to twenty years; as such, the base data model is very mature
 - The ACORD transaction model supports over one hundred different types of insurance transactions and is based on the data model
- Contrary to current beliefs, ACORD does have support for the Health LOB
 - It's missing support in some of areas of policy administration (as compared to the Life and Annuity LOBs), but insurance application/new business support looks relatively strong
- ACORD has the capability of being used as an enterprise wide insurance data format for Life and Annuity LOBs
 - Health support is being strengthened with each new semi-annual release
- While complex, ACORD has to be learned only once
 - Without a comprehensive, consistent standard like ACORD, every system-to-system wire format has to be learned individually

Pros - Technical

- ACORD is XML-based
 - This makes it possible to use many readily available, off the shelf tools for data transformation and manipulation
- ACORD has a data model extension mechanism that allows the addition of any fields that any system needs, but that ACORD does not explicitly specify
- Reusability becomes much more feasible when using a consistent data/transaction model foundation
- ACORD will encourage the creation of loosely coupled interfaces
 - This should reduce maintenance costs by increasing flexibility
- The ACORD data model will be very stable for a long time to come
 - Backwards compatibility is the central design rule for the model
- The ACORD schema directly describes the business data entities
 - This should make validation of ACORD datasets much simpler and more straightforward

Risks

- ACORD is very comprehensive and complex
 - The initial learning curve will be challenging
- The initial implementation times will take longer for the first interfaces
 - Mapping design initially will take longer due to the learning curve, but should decrease once more familiarity with ACORD is achieved
 - Development time will take longer initially, but should decrease once more familiarity with ACORD is achieved
- ACORD, like any other enterprise-wide standard, works best when it is centrally managed
- ACORD, like any other XML based protocol, can be very processor and memory intensive
 - This can lead to performance and scalability problems when processing ACORD-based messages

ACORD Data Model

ACORD Life Standard

- **Three core specifications:**
 - **TXLife – Life Business Messages**
 - **XTbML – Tabular Data Model**
 - **XMLife – Life Object Model & Data Dictionary**

ACORD Basic XML Concepts

■ Dates

✓ `<Birthday>YYYY-MM-DD</Birthday>`

■ Times

✓ `<SubmissionTime>HH:MM:SS</SubmissionTime>`

✓ `<SubmissionTime>HH:MM:SS-hh:mm
</SubmissionTime>`

■ Type Codes (Lookup Values)

✓ `<MarStat tc="1">Married</MarStat>`

Problems being solved: Life Data Model

■ Standard Data Exchange

■ Agreement on Common Structures:

- ✓ Vocabulary (Life Data Dictionary)
- ✓ Structure (Life Object Model)
- ✓ Support Unique Requirements in Standard Way
- ✓ Eliminate Custom Data Feeds

Transaction Architecture Support (TXLife)

Transaction Architecture Support (TXLife)

- **Request and Response Processing**
 - **User Authentication**
 - **Message Identification**
 - **Request/Response Data Requirements**
 - **Error Handling**
 - **Trial submits**
 - **Synchronous and Asynchronous**
 - **Updates or cancels of Asynch requests**
 - **Fire and Forget Functionality**
- *As other transport/transaction standards emerge much of this will move to the transaction wrapper layer – e.g. SOAP*

Life Object Model & Data Dictionary (XMLife)

Primary Objects in the Data Model

- **SourceInfo**
- **Activity**
- **Grouping**
- **Holding**
- **InvestProduct**
- **Party**
- **PolicyProduct**
- **Relation**
- **Currency**
- **FinancialStatement**
- **Scenario**
- **FormInstance**
- **SettlementInfo**
- **DistributionAgreement**
- **CommSchedule**
- **SystemMessage**
- **Campaign**
- **Criteria**
- **Case**
- **CodeList**
- **BusinessProcessDef**
- **AuditEntry**
- **UnderwritingGuidelines**

An ACORD Strategy

- Consider using ACORD for new interfaces or interfaces that are going to be re-engineered
- Do not change existing interfaces to use ACORD unless there is a compelling reason to do so
 - If an interface is being significantly rewritten, for example, then it may make sense to use ACORD
- Use the ACORD model primarily for application integration efforts
 - We will work with trading partners when they are ready to pursue ACORD
- Strong ACORD Governance Model – The ACORD canonical data model is an enterprise asset
 - All mapping done in conjunction with ACORD Data Architect
 - All extensions controlled by ACORD Data Architect
- ACORD Governance to be part of overall SOA Governance framework and SOA Governance tools.

Final Thoughts

- ACORD's main value proposition is in its ability to improve long term operational efficiency and effectiveness
 - The consistent, comprehensive foundation ACORD provides can be built upon in many ways
 - With consistency, wide-scale reuse becomes a real possibility
 - With comprehensiveness, predicting business needs becomes another real possibility
 - However, this foundation will not be implemented overnight
 - ACORD is a key part of the Consec IT strategy