

Deciphering the ACORD XML Standards Finding the Model Within

Tana Sabatino, Vallue Consulting

An Embarcadero Sponsored Whitepaper

November 2010

Americas Headquarters

100 California Street, 12th Floor
San Francisco, California 94111

EMEA Headquarters

York House
18 York Road
Maidenhead, Berkshire
SL6 1SF, United Kingdom

Asia-Pacific Headquarters

L7. 313 La Trobe Street
Melbourne VIC 3000
Australia

ABSTRACT

When implementing the ACORD XML standards for the insurance industry, one of the greatest challenges faced by companies is finding the data model inside of them. Depending on the line of business and standard you need to support, the data model may be published as such only for you to realize it isn't a true model at all but rather a data hierarchy. Or it may simply be embedded inside of a message structure with no back up documentation on how the data is modeled. Or a logical model is published which bears little resemblance to the physical XML structure.

However, in order to read and write to the standards and marry the data with your systems, it is imperative that you understand how the information is modeled. Otherwise, there's no knowledge of how data relates to one another making implementation impractical at best. No one anticipates that any one company, system, interface, or database would match these XML structures. Yet companies need to fully understand them in order to successfully consume the data.

This paper delves into the technical aspects of finding the data model within the ACORD standards. It will describe what to look for and how to extract those structures out so that it can be represented in a data model that provides greater understanding and eases implementations for companies trying to maximize the benefits of these industry assets.

INTRODUCTION: XML AND DATA MODELING - THE CORRELATION

XML standards are prevalent across the insurance marketplace for communication and data exchange. The benefits of utilizing standards are obvious to industry players. Insurance companies recognize that they do not operate in isolation nor do they have the massive IT budgets of the 1980s that afforded them the ability to go it alone. To survive in this industry, it is imperative that companies can communicate with multiple trading partners including brokerages, service providers or simply the end client. Recognizing that the tools available in the marketplace are superior and more cost effective than anything that can be developed in-house further justifies the need for standards. Even for internal interface development efforts, companies are recognizing that they are far better off using an XML standard that was designed with thousands upon thousands of man hours with input across many companies than trying to develop something internally themselves.

Nevertheless companies still struggle. The standard is not written in their native tongue. Their internal systems use a different vocabulary and structure than the format of the standard. In fact, each internal system within a single company will use a different vocabulary and structure compounding the issue around comprehension. In essence, each system and department speaks a different language and now one more is being imposed for communication purposes.

One ACORD user describes the ACORD XML standards as the insurance company's equivalent to "Esperanto." Esperanto is the most widely spoken constructed international auxiliary language. In the late 1880s, the goal of its creator, L.L. Zamenhof, was to develop an easy-to-learn and politically neutral language that would serve as a universal second language to foster peace and international understanding. It was considered a *second* language; it was never a goal that it be any country's primary language.

While today we are not all expert users of Esperanto, its purpose is akin to what the ACORD XML standards serve for the insurance industry - a politically neutral language that serves as a universal second language to foster communication between systems and trading partners.

Still companies struggle with its adoption. The biggest issue? Unlike Esperanto, the structure is not easy to learn. On the surface it seems straightforward. After all, it is an XML vocabulary. It has named tags, pre-defined lists of values, and an organized structure. How complex can it really be? Companies discover that once they get beyond the surface, understanding that organized structure becomes the challenge. And an even greater challenge is correlating that structure to each and every system's internal structure so that the information can be mapped and shared correctly.

Where XML fails in general is being able to show inherently the logic and modeling behind the structure that is ultimately communicated.

We speak of XML vocabularies. People often assume an XML vocabulary is nothing more than a data dictionary. But this is wholly inaccurate. A vocabulary involves both the words in the dictionary as well as how they are structured and related to one another so that they make sense.

While we may no longer remember third grade, anyone with a school age child is quickly refreshed on the art and science of diagramming sentences. Sentences can be very complex, and can contain many different parts of speech which implicate many different grammatical rules. Structure a sentence incorrectly and the meaning can change dramatically. For instance, consider the following examples:

Only the new menu confused the customer.

The new menu only confused the customer.

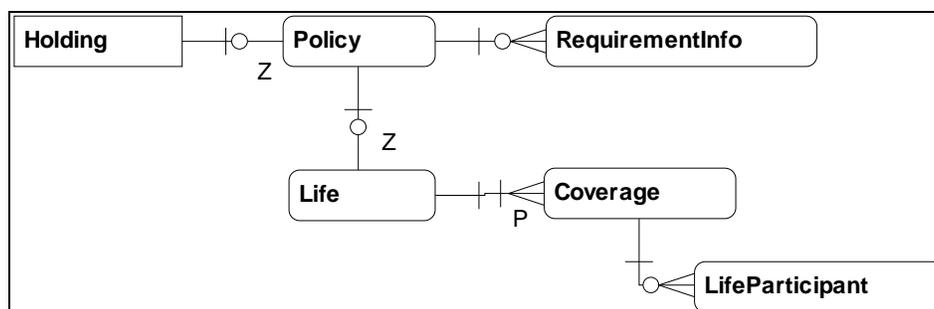
The new menu confused only the customer.
The new menu confused the only customer.

Moving just one word changes the entire meaning. In the data world, sentence diagramming is akin to data modeling. Data modeling is nothing more than defining the data elements and the relationships between them. In order to develop XML standards that can be used consistently in a cost effective manner, data modeling must happen. Even though XML is a hierarchical layout, all ACORD XML standards have done data modeling in order to achieve that hierarchical layout.

MODELING APPROACHES WITHIN THE ACORD STANDARDS

Each standards initiative within ACORD refers to the data model and their modeling efforts differently. But the reality in all cases is that modeling was done.

For the LAH (Life, Annuity, and Health) standards, creation of a data model is an explicitly stated objective. This model supports life insurance, disability insurance, annuities, long term care and health insurance products. It covers the information needed for a wide variety of business processes, from product profiling and new business, to commission payments, policy administration and claims settlements. This model is published as an object hierarchy diagram, within an XML schema that directly reflects this structure. Below is a sample of what a snippet of the object hierarchy looks like when shifted into a traditional entity-relation model.

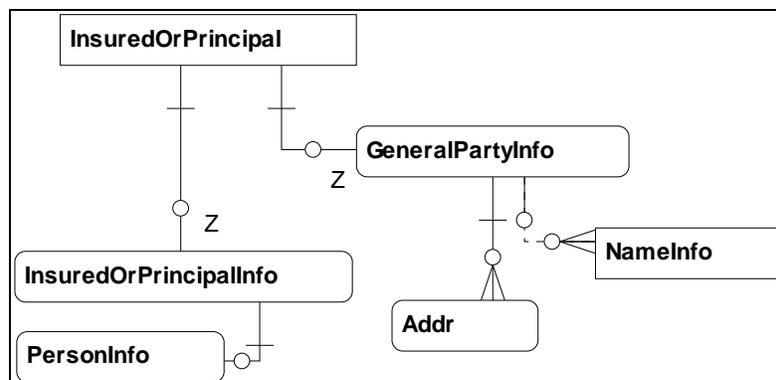


Snippet of ACORD Life Model, from Embarcadero's ER/Studio

In the P&C industry, while a data model is not a stated objective of the standards participants, they nonetheless spend a significant amount of time modeling data in order to normalize information and represent concepts consistently across messages. This standard addresses the needs of a large number of product lines for both personal and commercial lines, particularly around the sharing of policy and party information in support of new business and renewal processing. Since the P&C industry has shied

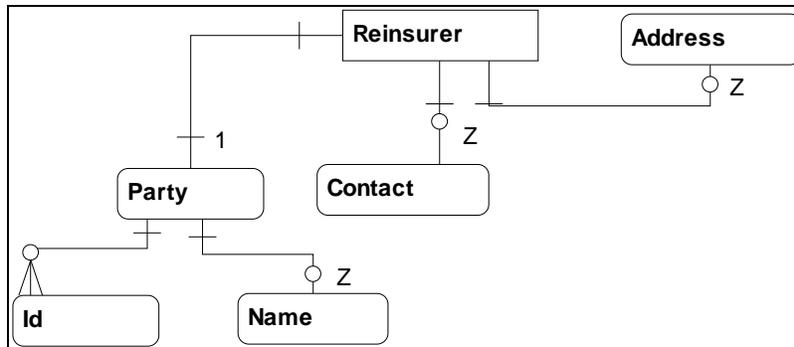
away from describing their effort as a 'data modeling' effort, finding the 'data model' is a bit more challenging than that of the life industry because they do not publish the model as a picture.

Thus to find the actual model, you have to identify the line of business you want and select a message for that line of business. By investigating the contents of that message, it becomes obvious that there are parties, policies, coverages, relationships, etc. This is the model. And if you look across the messages, you'll see that the vast majority all use the exact same aggregates with the line of business aggregates being the sole differentiator. So while it may not be called a 'model', there is definitely one in there. Below is a snippet of what the P&C XML structure looks like when shifted into a traditional entity-relationship model.



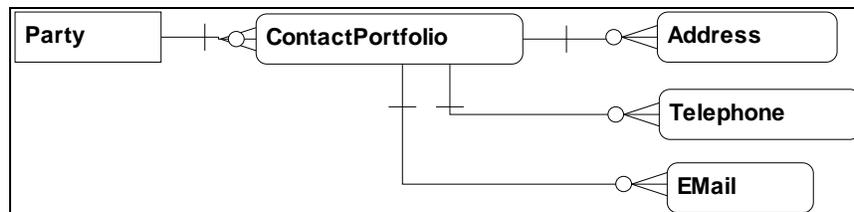
Snippet of ACORD P&C Implied Model, from Embarcadero's ER/Studio

ACORD also has a standard for the property and casualty reinsurance and large commercial industry. This standard, similar to the life standard, states that having a data model is an objective of the standard. Where it differs from ACORD Life, however, is that the data model itself is not represented in its native form in the XML messages. Thus, to find this data model, one utilizes the documentation database provided by ACORD. The interface itself demonstrates the model as well as provides mappings to the XML structures. This provides a level of abstraction between the data model and the XML structure itself.



Snippet of ACORD P&C Reinsurance Model, from Embarcadero's ER/Studio

Lastly, anyone active within ACORD has been hearing a lot about ACORD's current efforts to create a unified model across lines of business. Referred to as the ACORD Framework with the ACORD Information Model being one component, it offers promise to tie together the different ACORD standards so that they can be related to one another: an Esperanto for ACORD XML if you will. It is not expected that this model will necessarily be reflected in an XML message. It is still under development and its ultimate role is still being determined. UML is the modeling language of choice for this effort. Below is a snippet of what this looks like.



Snippet of ACORD Information Model 2.0, from Embarcadero's ER/Studio

The greatest complaint across all these standards is if and how the model is published or even found. Even for the life insurance standard where the model is explicitly created and published, it is not a traditional entity-relation model but a model hierarchy that fits the published XML structure. For the other standards it becomes even more difficult to figure them out.

Companies must review these standards differently than just looking at the XML structure and documentation provided by ACORD. If it were that simple, companies would not be struggling. So what do we do?

FINDING THE MODEL WITHIN

Each standard starts at a different point when it comes to data modeling. Yet for every case, some very basic principles apply.

CORRELATE LIKE PRINCIPLES

When there is any level of data modeling taking place for creating XML messages, one ends up with at minimum hierarchical messages with elements grouped (called aggregation) with these aggregates repeating when logical. To correlate this structure to a data model, the following simplistic rules apply:

An XML aggregate correlates to a data model entity

The XML elements within the aggregate correlate to entity attributes

An aggregate that is nested inside another aggregate to build a hierarchy becomes a relationship between entities in a data model

This is the most simplistic take on converting XML to a data model. Items 1-3 are usually handled by tools that import XML schemas automatically, such as ER/Studio's Metadata wizard. It is really all tools can do automated because they do not have knowledge of the business content inside. The end result is a nice starter but still isn't what one hopes for. So what's next?

KNOW WHAT'S IN THERE

ACORD provides some great reference material for these standards. In order to find the model within the standard, the basic structure, supported lines of business and business processes needs to be understood. This does not require a modeling effort but just a basic understanding of the XML structure and the scope of the information within. Without a basic understanding, the rest of these points are moot.

Mind you, this is critical for the people who are doing the modeling effort. Once the model is done, then all users of the standard within your enterprise will have the model asset to use as a starting point. Some users, such as your business analysts and data mappers may never even have to see the XML standard itself.

DETERMINE WHERE THE BUSINESS DATA STARTS

In the ACORD XML standards, transaction information is combined with business data in order to create an XML message. From a data modeling perspective, the transaction instructions become irrelevant. Find where the business data section(s) begin and work from there.

ACORD Life Message

```
<TXLife Version="2.23.00">
```

```
<TXLifeRequest>
```

```
<TransRefGUID/>
```

```
<TransType tc="502"/>
```

```
<TransExeDate/>
```

```
<TransExeTime/>
```

```
<OLifE>
```

```
<Holding>
```

```
...
```

ACORD P&C Message

```
<ACORD>
```

```
<SignonRq/>
```

```
<InsuranceSvcRq>
```

```
<RqUID/>
```

```
<DwellFirePolicyAddRq>
```

```
<RqUID/>
```

```
<TransactionRequestDt/>
```

```
<CurCd/>
```

```
<InsuredOrPrincipal/>
```

```
<PersPolicy/>
```

```
...
```

**Aggregates and elements are shown as empty solely for illustrative purposes.*

BREAK IT DOWN

The ACORD standards are large and cover a broad range of business lines and business processes. Determine the area(s) that matter for your implementations and split off the rest. If you only sell personal lines, remove the commercial lines structures. If you will not be using the standard for product profiling, ignore that area of the XML standard for now. Focus on the critical areas first such as parties and the lines of business you support to build a comprehensive model.

LOOK FOR PATTERNS

In each of the standards, a pattern was used in the definition that can be extrapolated for developing a model. For instance, in the property and casualty standard, the aggregate that represents various party roles include two main sets of information:

generic party data and a separate aggregation that defines data specific to its role in an insurance contract. This type of pattern, once uncovered, will lend itself to being able to build tables and relationships consistently across the various entities within the standard.

```
ACORD P&C Message
<Producer>
  <GeneralPartyInfo>
    <NameInfo/>
    <Addr/>
    ...
  </GeneralPartyInfo>
  <ProducerInfo>
    <ContractNumber/>
    ...
  </ProducerInfo>
</Producer>
*Aggregates and elements are shown as empty
solely for illustrative purposes.
```

RESOLVE REFERENCES

XML uses XML IDs and IDREFs, which are data types specific to XML, to create the concept of primary and foreign keys. However, this implementation does not match an entity-relation interpretation of this same concept. The biggest issue is that IDREFs do not allow the reference point to be specifically named. It references any XML ID, even if the ACORD documentation and the name of the item specifically states it is a reference to a specific type of aggregate. The fact that it is a reference to a specific aggregate type is lost in XML itself and must be resolved for modeling purposes.

```
ACORD Life Message
...
<LifeParticipant PartyID="Beneficiary_1">
  <LifeParticipantRoleCode tc="7"/>
</LifeParticipant>
...
<Party id="Beneficiary_1">
...
*Aggregates and elements are shown as empty
solely for illustrative purposes.
```

In some cases, the resolution is not this straightforward. Because the XML IDREF construct can reference any XML ID, some portions of the standards use it in its generic nature and allow it to reference anything. For instance, in the ACORD Life Standard there is a generic Relation object that allows users to relate a set of objects to one

another using this concept. The resolution of the IDREFs must be taken one step further and mapped to the type of relationship to be able to properly build relation tables.

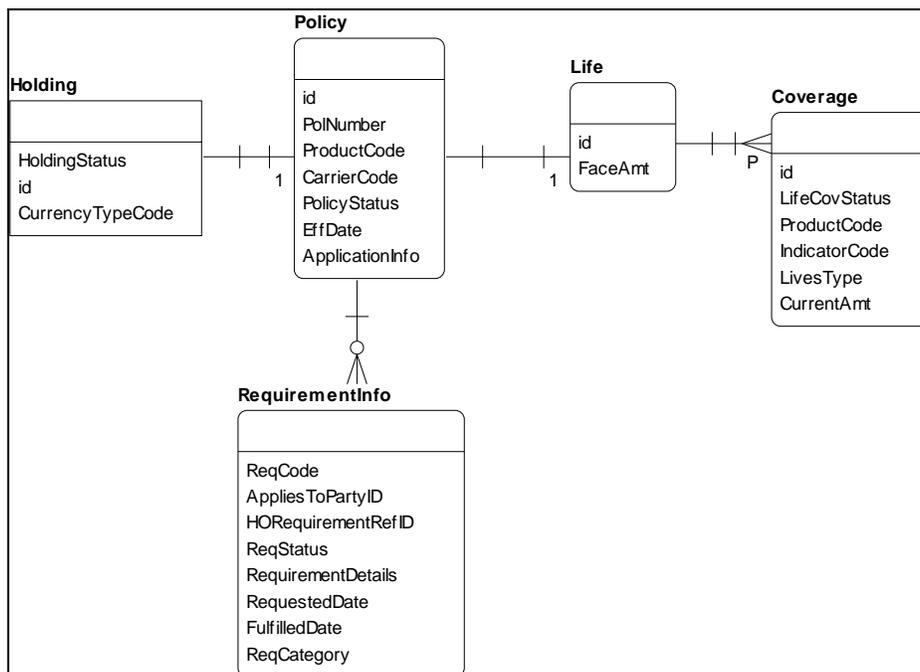
```

ACORD Life Message
...
<Relation
  OriginatingObjectID="Annuitant_1"
  RelatedObjectID="Beneficiary_1">
  <RelationRoleCode
    tc="2">Child</RelationRoleCode>
  </Relation>
<Party id="Annuitant_1"/>
<Party id="Beneficiary_1"/>
...
*Aggregates and elements are shown as empty solely for illustrative purposes.

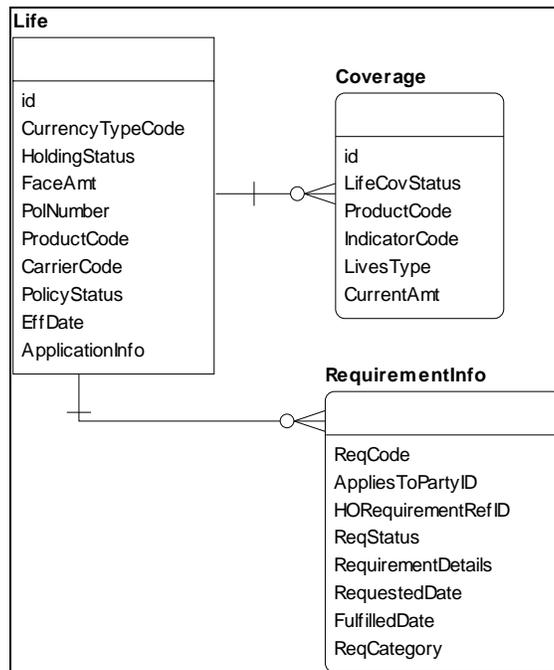
```

RESOLVE DIFFERENCES IN XML AGGREGATION VERSUS DATABASE ENTITIES

In addition, many aggregates constructed hierarchically make up a single entity concept. For instance, in the ACORD life standard, a life policy is comprised of data that is a 1-1 relationship between the Holding, Policy and Life aggregates. Companies must decide when to group these together as a single entity for modeling purposes or keep them separate relating them 1-1 to one another.



Before collapsing, from Embarcadero's ER/Studio



After collapsing, from Embarcadero's ER/Studio

This scenario should be used with caution on a case by case basis. The ultimate use of the data model needs to be considered before arbitrarily collapsing entities. If the primary goal of the model is to aid comprehension and mapping to/from the XML schemas, then collapsing entities makes this more difficult because the entities are not going to map 1-1 back to the schema. However, if the goal is to build a model that is based off the XML schema for use in a database environment, then this task may have greater applicability by reducing the number of tables and joins needed.

UTILIZE TOOLING

A picture is worth a thousand words. For data modeling, this cannot be more true. Too often people open up a schema, have puzzled looks on their faces and don't know what to do from there. Utilizing tooling to help navigate through the schema in model form and making the adjustments necessary for both their comprehension and reusability for other environments will serve companies very well. The modeling examples above are either directly from an XML schema or a direct import into [Embarcadero's ER/Studio](#). With a simple import of the schema using the Metadata Wizard, the schema is automatically converted into a data model. With this functionality, users can immediately visualize the model and adjust as needed. Tables can be merged; unused entities and attributes can be deleted; relationships can be explicitly defined.

More importantly, this new model can then be mapped to the back-end enterprise systems that must send and receive the XML data. Knowing where items are used and understanding their data lineage is now achievable by creating an inventory of user defined mappings between physical models within companies' internal systems and the ACORD standard in a data model form. This will allow companies to reap further benefits of reusability and change management across their systems' efforts.

SUMMARY

Implementing the ACORD XML standards are challenging. They are rich in content and complex as a result. To achieve the full benefit of the ACORD standard, finding the model is a necessity. By utilizing modeling tools to apply basic principles for breaking down the XML to discern the data model within the structures, companies will see a great advantage in being able to understand and utilize them. With a model, companies can correlate and map the information consistently between their internal systems and the XML standard. Without a model representation, companies will continue to struggle to understand how the standard works.

ABOUT TANA SABATINO

Tana is the founder of Vallue Consulting Inc. (www.vallue.com) which provides technology consulting for the insurance industry. Vallue Consulting specializes in data modeling, the application of XML and ACORD's insurance standards. She formerly led the insurance industry standards efforts at ACORD. Here, Tana was instrumental in ACORD's success in meeting the challenges of a continuously changing industry. During her tenure, she introduced and developed ACORD's life standards program, and steered the development efforts of ACORD's XML standards. Tana is known industry wide as an expert for insurance XML.

ABOUT ACORD

Based in New York, ACORD (Association for Cooperative Operations Research and Development) is a global, nonprofit insurance association whose mission is to facilitate the development and use of standards for the insurance, reinsurance and related financial services industries. With offices in London as well, ACORD accomplishes its mission by remaining an objective, independent advocate for sharing information among diverse platforms. ACORD Standards and services improve efficiency and expand market reach. Affiliated with ACORD are hundreds of insurance and reinsurance companies, and thousands of agents and brokers, related financial services organizations, software providers, and industry organizations worldwide



Embarcadero Technologies, Inc. is the leading provider of software tools that empower application developers and data management professionals to design, build, and run applications and databases more efficiently in heterogeneous IT environments. Over 90 of the Fortune 100 and an active community of more than three million users worldwide rely on Embarcadero's award-winning products to optimize costs, streamline compliance, and accelerate development and innovation. Founded in 1993, Embarcadero is headquartered in San Francisco with offices located around the world. Embarcadero is online at www.embarcadero.com.