



Enterprise Technology & Architecture Services

Implementing a Federated Canonical Model

Presented By:

Doug Stacey

May 8, 2011



Overview



-
- Allstate Company Overview
- Strategy – A Federated Approach
- Allstate Canonical Model
- Structuring the Model In IgniteXML
- Mapping
- The Service Architecture Review Group
- Naming Conventions
- Vocabulary and the Allstate Data Guide
- Next Steps...



- ♣ **The nation's largest publicly held personal lines insurer of homes and autos**
- ♣ **A Fortune 100 company with \$157.5 billion in assets**
- ♣ **Allstate sells 13 major lines of insurance, including auto, property, life and commercial. Allstate also offers retirement and investment products and banking services**
- ♣ **Allstate is widely known through the “You’re In Good Hands With Allstate®” slogan**
- ♣ **The Allstate Corporation encompasses more than 70,000 professionals with technology operations located around the globe**
- ♣ **More than 17 million customers in the U.S. and Canada**

Overview - Enterprise Integration Modeling at Allstate



- The objective of federated integration modeling is to provide a dictionary of common objects and definitions per domain level to enhance system interoperability
- This strategy aims to develop a federated repository containing a suite of XML Schema based message sets
- It will develop the infrastructure to support the Allstate Canonical Model

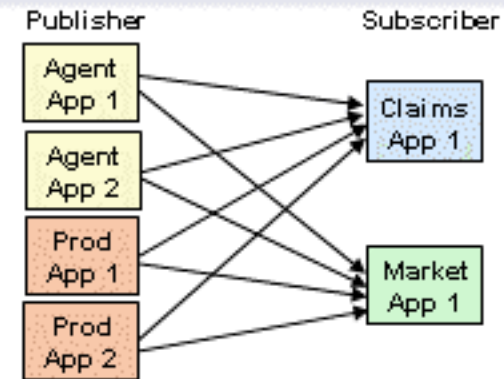
Infrastructure = Standards or Allstate Bespoke +
Tool Support (igniteXML, et al) +
Implementation Strategy

Canonical Model Options



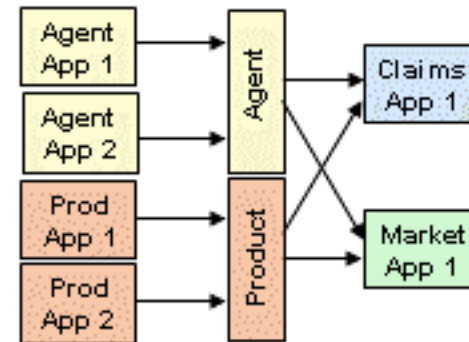
Federated by SoR

- Federated by SoR approach is closely tied to the System of Record.



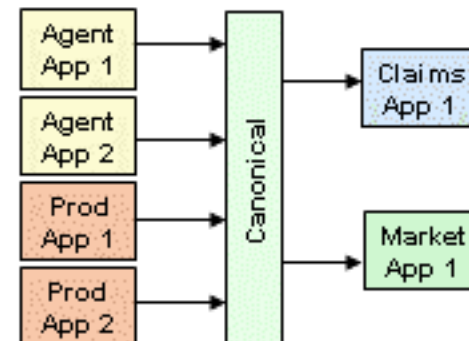
Canonical by Business Domain

- Canonical by Business Domain approach is closely tied to core business function (Policy Administration, Claims, etc.)



Canonical

- A common or shared set of structures that are used by all systems to exchange information





Why a federated approach over a single canonical model?

Pragmatism!

Allstate Canonical Models



- The Allstate Canonical Model is a reusable resource that Information Architecture will use to build their application specific messages

- The individual message sets are derived from the appropriate Domain Canonical Model

- This approach improves the opportunities for :
 - Governance,
 - Reuse,
 - Consistency and
 - Interoperability

Rules for a Federated Approach



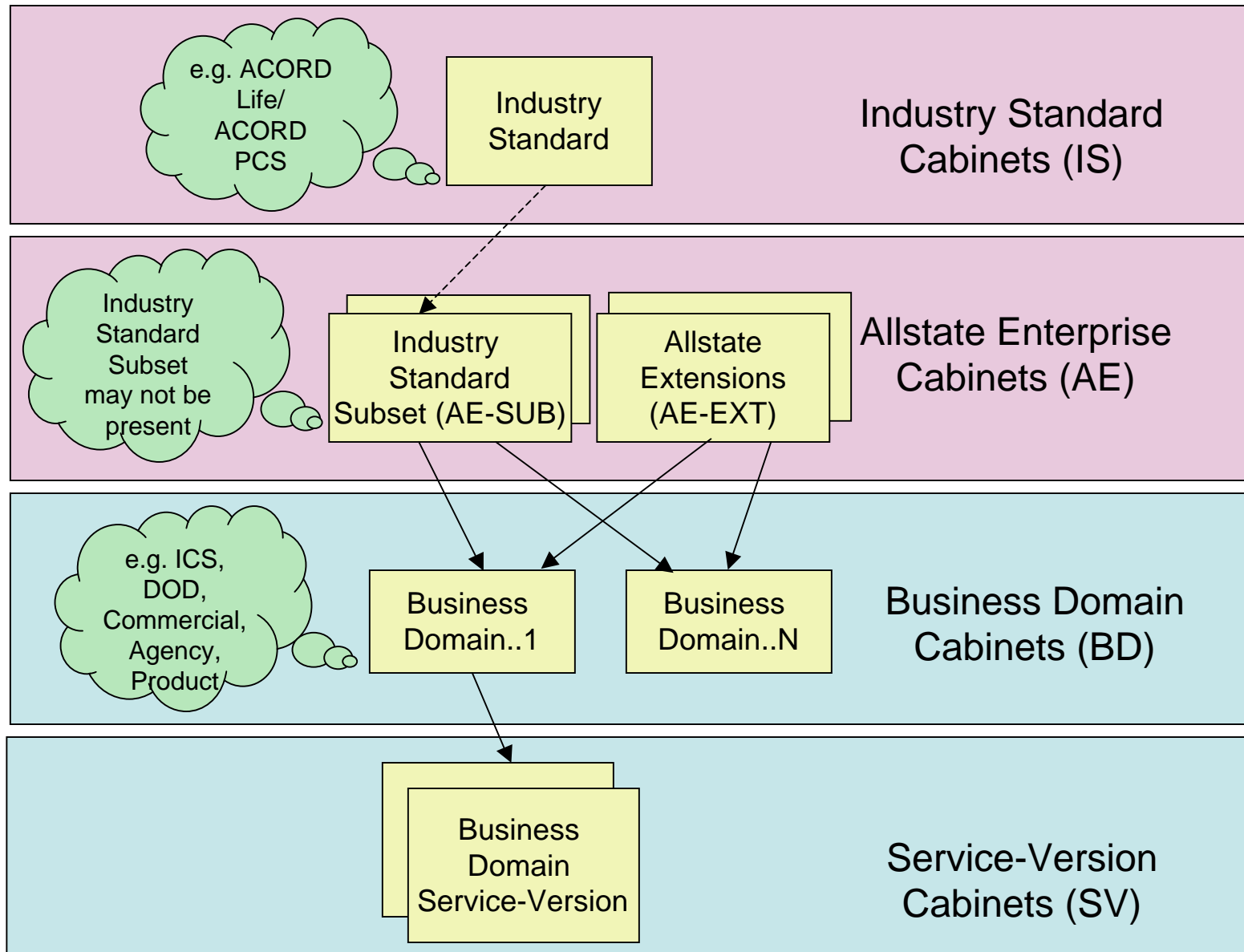
- Canonical by Business Domain approach implies that each business domain will use its own canonical model insuring consistent representation of business domain data across the enterprise
- Sourcing of data will require the System of Record declarations to be in place identifying what data is owned by individual SoR with orchestrated services providing access to Business Domain data that resides across multiple SoRs that are owned by the Business Domain
- Information Architecture, in collaboration with the rest of Architecture Services will set the syntactical framework (XML naming rules, data types, design patterns etc.)
- Business Domains will implement its services by applying the syntactical framework to the content they own based on the SoR declarations. The service interface design work would be done under the oversight of Information Architecture/DAG and Data Management
- When a system has to expose data that is owned by a different Business Domain it will align with the semantics and syntax that was established by the owning Business Domain
- Service interfaces based on the Canonical by Business Domain approach will be built by Business Domains once the underlying SoR are declared and syntactical framework is available (published)
- Implementation of the Canonical by Business Domain Model across Domain applications will be managed to insure consistency within and across Domains. When several Business Domains elect to implement a shared model (ACORD) extensions to the model will be centrally managed to insure consistency and reuse. The exact governance model will be determined by Information Architecture in collaboration with the implementing domains.

Scope for a Federated Approach



- This direction applies to A2A cross domain integrations at Allstate and is primarily focused on integrations based on XML.
- New applications and applications working through remediation or upgrade efforts must remediate to apply the canonical model to their cross domain service interfaces.
- This position statement impacts every Enterprise/Cross Domain Service design.
- This position does not apply to B2B integrations where use of industry standards is recommended.
- The use of canonical domain model is not mandated for intra domain integrations.

Structuring the Model – A Layered Approach



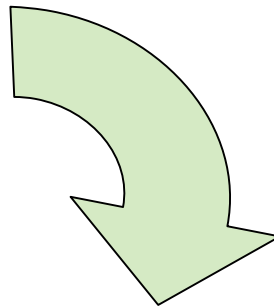
Transition Plan – Building the model



- The initial phases of the project involve harmonization of existing schemas into the Allstate Canonical Model

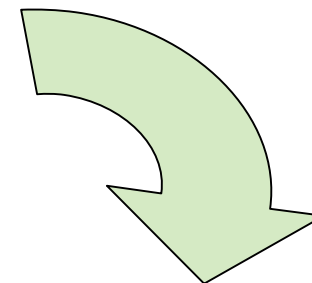
1st Iteration ACORD LIFE

1. ICS 1x and 2x
2. AGENCY 1x and 2x
3. DOD
4. EMPLOYEE



2nd Iteration PC&S

1. ABI (Commercial)
2. SERVICE BOSS
3. PRODUCT



3rd Iteration NEXTGEN

1. CLAIMS

Tool Support – For Schema Management



- Historically - file based approach using XMLSpy
- Currently transitioning to igniteXML Organizer
- Backlog of Schemas to include - Rough Order of Magnitude (ROM) Schema Sizes

<u>Schema</u>	<u>Elements</u>
CUSTOMER	600
AGENCY	400
DOD	700
EMPLOYEE	200
PRODUCT	3000
CLAIMS	??

Tool Support – For Mappings



- Integration Projects developing XML Schema based messages at Allstate typically include the following activities:
 - Determining Interface Requirements
 - Mapping Interface Requirements to Schema Elements
 - Building and Extending the Canonical Model to Support the Mappings
- For the Mappings we have two tool usage approaches:
 1. Excel when we have a single resource doing the Mappings
 2. DataLink Mappings Manager for team based Mappings
- Both provide the ability to manage mappings using:
 - *From* → *To* fields
 - *From Description* field
 - *Transformation* field



DataLink Mappings Manager

[Home](#) | [Hide / Show Menu](#) | [Tools](#) | [Contact Us](#)

Menu

Admin

- [Project Folders](#)
- [Databases](#)

Manage

- [Systems](#)
- [Mapping Groups](#)
- [Users](#)

Export

Help

Manage Mapping Rows

DataLink Mapper

Back Links: [DataLink Mapper](#) |

Mapping Project :

Group Name :

Filter By :



Limit Source Columns To :

Select

Export

From	To	From Desc	Transformation	Operation
				Map Delete
				Map Delete



SARG will facilitate Services/Events consistency, reuse and quality by governing adherence to SOA- and EDA-related Principles, Directions, and Standards to projects and systems

● **SARG expects to achieve this by:**

- Reviewing and approving requests for service creation, consumption and event registration
- Shaping Service Contracts and Event Classifications for optimal reusability
- Promoting service and event reuse

● **Governance Priorities**

- Optimize service granularity
- Alignment with common (canonical) schemas including namespaces
- Use of enterprise codes (e.g., Event Type)
- Use of standard headers and properties

● **SARG does not fund development or maintenance of services, but it will assist in finding the best owner for services and work with the development teams to design highly reusable services and events.**

Naming Conventions - Cabinets



● General

- *lower-case-with-hyphens-for-spaces*

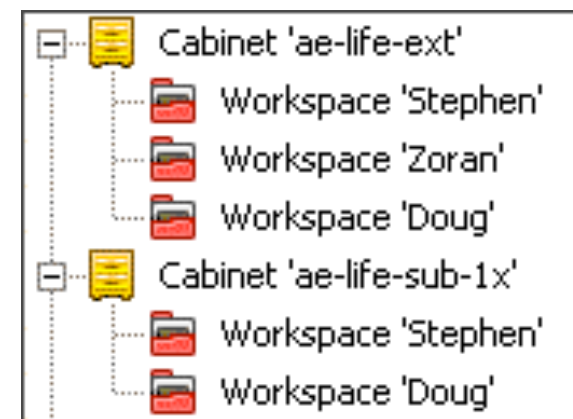
● Cabinet

- is-<industry-standard-name>-<industry-standard-version>
- Example is-life-2.23.00

- ae-<allstate-enterprise-cabinet-name>-<ext | sub>(-<version>)
- Example ae-life-ext

- bd-<business-domain-cabinet-name>-<version>
- Example bd-ics-2x

- sv-<service-version-cabinet-name>-<version>
- Example sv-agency-locator-1x

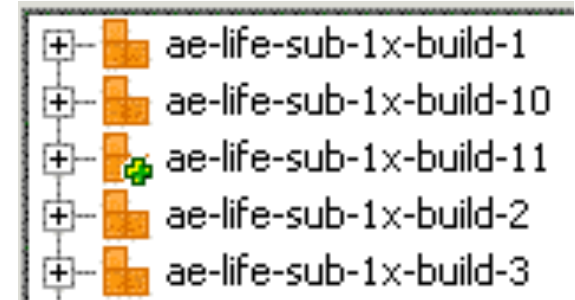


Naming Conventions – Builds and Releases



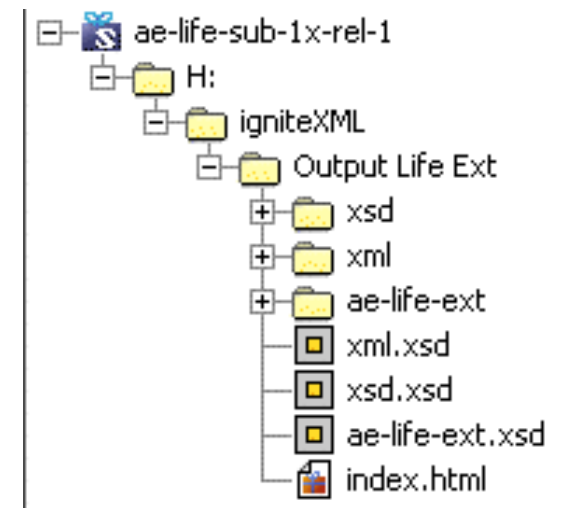
Build

- `<is | ae | bd | sv>-<-cabinet-name>(-<version>)-build-<build-number>`
- Examples
 - *is-life-2.23.00-build-1*
 - *ae-life-ext-build-25*
 - *ae-life-sub-1x-build-39*



Release

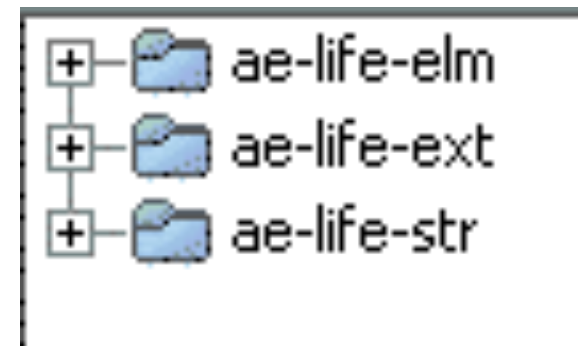
- `<is | ae | bd | sv>-<-cabinet-name>(-<version>)-rel-<release-number>`
- Examples
 - *is-life-2.23.00-rel-1*
 - *ae-life-ext-rel-25*
 - *ae-life-sub-1x-rel-39*



Naming Conventions - Core Components



- Core Component
 - **ACORD Elements**
 - *ae-<allstate-enterprise-cabinet-name>-elm*
 - *Example ae-life-elm*
 - **Structures**
 - *ae-<allstate-enterprise-cabinet-name>-str*
 - *Example ae-life-str*
 - **Allstate Extensions**
 - *ae-<allstate-enterprise-cabinet-name>-ext*
 - *Example ae-life-ext*
 - **Messages (Service Version Cabinets Only)**
 - *ae- <allstate-enterprise-cabinet-name>-msg*
 - *Example ae-life-msg*



Data Dictionary Management - Allstate Data Guide



- At Allstate data definitions are managed using the Allstate Data Guide
- Current plan is to import the definitions into igniteXML using the Vocabulary feature
- Looking at efficient ways to keep the two systems in sync

PARTY

Details

Attributes

Related Entities

Keys

Associated Files

Definition : A Party is a person, institution, or group.

Model Name : [Integrated Customer Services](#)

Entities with the same name also appear in the following data model(s):

[Agency Gateway Content](#)

[Agents Selection Process](#)

[Billing](#)

[Branch Office Agent Table / BOAT Declaration](#)

[CANVAS](#)

[CASE File](#)

[Claim Stats](#)

[Claims History Repository](#)

[Comparative Rating System](#)

[Customer Database / CDB](#)

[Data Solutions](#)

[Enterprise Campaign Management / ECM Mart](#)

[Enterprise Event Handler](#)

[Enterprise Logical Data Model / ELDM](#)

[Household](#)

[In Market Pilot Campaign Database](#)

[Integrated Risk Management System / IRMS](#)

Vocabulary in igniteXML



igniteXML 5.2.1 - Cabinet: Enterprise Information - User: smeri Oracle

Cabinet Edit View Tools Help

Vocabulary | Meta Data

Browse vocabulary and taxonomy definitions

Life
└─ Party

Party

Taxonomy Properties

Name: Party

Qualified Name: Life.Party

Business Identifier:

Definition: Parties are People or Organizations.

Detail: Producers, Staff, Agency Corporations, Emergency Contacts, Field Leaders are all types of Parties.

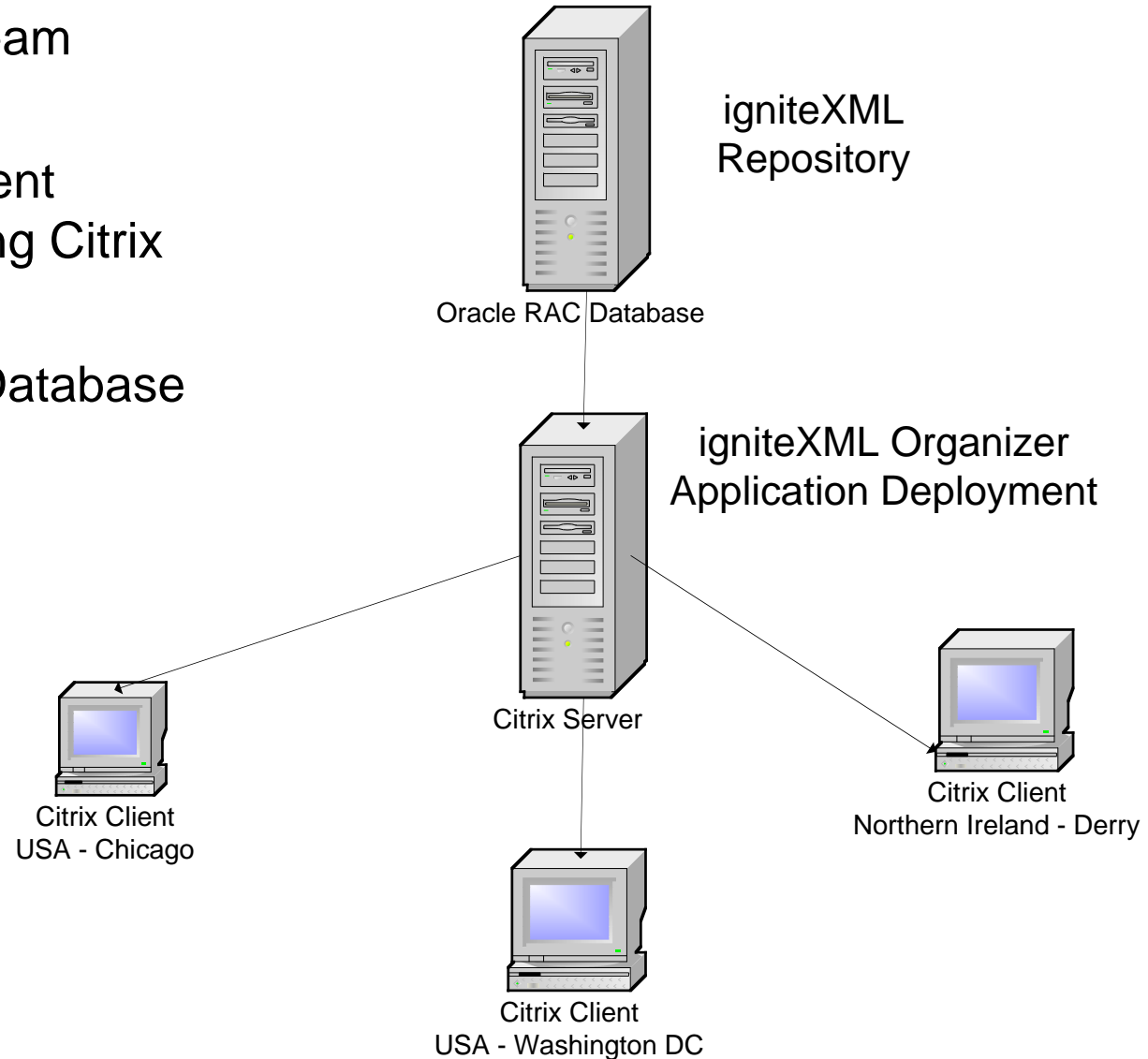
Deprecated:

Type: Vocabulary

igniteXML Deployment Environment



- Distributed Team
- igniteXML Client Deployed using Citrix
- Oracle RAC Database



Plans for the future...



- Upgrade to igniteXML Organiser version 5.5.1
- Build out the Allstate PCS based Canonical Model
- Review “First in Wins” Issue – More harmonization required in 2x
- Review multi-level Cabinet Structure
- Research and formulate strategies for:
 - Vocabulary
 - Meta-data
 - Annotations
- Complete an igniteXML Implementation Review