AUTOMOTIVE INDUSTRY GUIDELINE FOR DIRECTORY SERVICES

Automotive Industry Guideline for Directory Services	**AiAG**

AIAG PUBLICATIONS

An AIAG publication reflects a consensus of those substantially concerned with its scope and provisions. An AIAG publication is intended as a guide to aid the manufacturer, the consumer and the general public. The existence of an AIAG publication does not in any respect preclude anyone from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the publication.

CAUTIONARY NOTICE

AIAG publications are subject to periodic review and users are cautioned to obtain the latest editions.

MAINTENANCE PROCEDURE

Recognizing that this AIAG publication may not cover all circumstances, AIAG has established a maintenance procedure. Please refer to the Maintenance Request Form at the back of this document to submit a request.

APPROVAL STATUS

This document was approved for publication by the AIAG Board of Directors on July 31, 2001.

FOREWORD

The Electronic Commerce (EC) Directory Services Work Group has developed a guideline for structuring a directory schema to present information about People, Companies and Locations to a community of trading partners. This guideline uses a standard mechanism, the Lightweight Directory Access Protocol (LDAP), to implement this schema. This schema includes the most common information trading partners need to know about one another to do business on a day to day basis, as well as a procedure to extend the schema either as a private extension or an extension to the AIAG guideline.

A common schema used by all trading partners will lower overall industry costs for publishing and facilitate obtaining information. Queries made from one trading partner to another will be simplified. The LDAP protocol is implemented in many common e-mail and web browser packages. Most companies will not need to acquire custom client software. Server software is also widely available and can be implemented either by individual companies to serve their own needs or by service providers who can then offer their services to many clients.

In summary, this guideline defines a demonstrated methodology that allows all automotive industry trading partners to implement common directory schema across a global and diverse business community. The net effect is to reduce proliferation of purchased custom software. The result is reduced costs to the automotive industry in general.

ACKNOWLEDGEMENT

Automotive Industry Guideline for Directory Services

TABLE OF CONTENTS

1.0 INTRODUCTION

Collecting, publishing, and accessing information about people and resources is a human need that dates back to the beginning of civilization. Historically, the kinds of information that have been collected include government and organizational hierarchies and geographical and locality information, as well as ways for people and resources to find each other. This information is represented in publishing mechanisms known as directories. The success of technologies like the telephone system is based on the easy availability of this information. Given that telephones have been around for the last century, you can see how directories have become a key to our ability to communicate.

Manufacturers using networked computing resources today face a major directory challenge, whether they use the Internet or other communications technologies. As existing relationships expand and new ones develop, the need to share common business information among organizations is obvious. This information includes points of contact and security information, as well as the kinds of products and services that a manufacturer provides. With the successful deployment of the ANX®, the need for automotive participants to publish and locate this information is becoming critical.

The application that enables this publishing of directory information is known as a directory service. Within organizations, a number of directories exist. They are embedded in applications like electronic mail and network operating systems. Some organizations have even developed enterprise directories to tie this information together. However, within the industry and between organizations there are relatively few shared directories and fewer still that enable commerce over the Internet or ANX®. Without directory services, we are limited in our ability to locate or share information and, by extension, to conduct business as efficiently as we might.

Directory services consist of an information store and the communications protocols that provide access to the data maintained in that store. Industry standards for directory services include the International Telecommunications Union X.500 standards and the Internet Engineering Task Force Lightweight Directory Access Protocol (LDAP). Standards-based solutions today incorporate the X.500 information model and LDAP access mechanisms. Every major software provider, including IBM, Netscape, Microsoft, and Novell, is adopting these standards.

From an industry perspective, a directory service represents an information base that is contributed to by industry participants. Shared information may be extracted from existing corporate directories or it may be submitted directly by a user. Strategies to model, collect, maintain, and control access to this information will need to be developed. In particular, a Directory Service may be a major element of a Public Key Infrastructure (PKI). The PKI allows suppliers to share digital certificates used to sign electronic correspondence and to encrypt information. Industry policies for practices such as issuance, validation, and revocation of public keys must be established to support the levels of organizational trust that will be associated with digital certificates. These capabilities are a basic component for conducting electronic commerce in the future.

Figure 1.

Company
Directory

Internet
or other
network

Directory
Service
Provider

Company
Directory

Application
Server
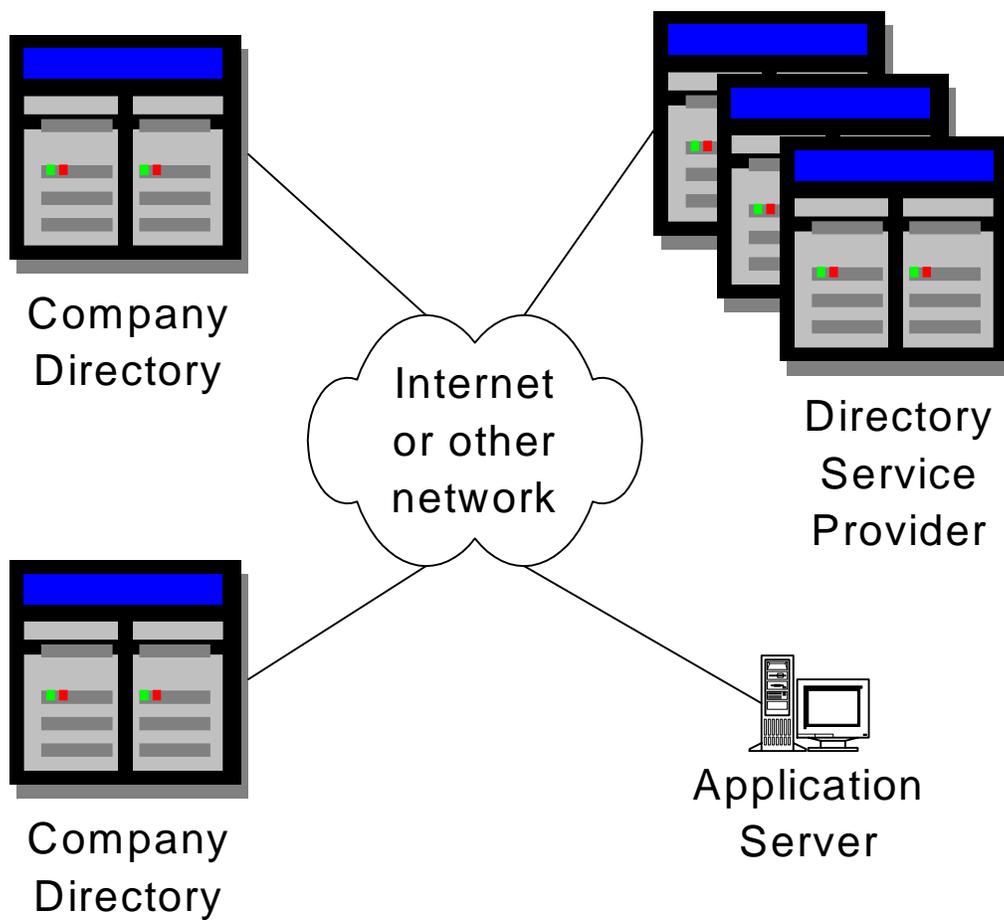
2.0 THE BUSINESS CASE

In today's computing environment, the value of a corporate network is measured more and more by how well it exploits the global Internet. There are two facets to making effective use of the intranet-Internet link:

- Making the rich resources of the Internet readily available to corporate users, enabling them to gather information and to communicate with customers, suppliers, and other trading partners.

- Employing open, scaleable Internet technologies (e.g., ANX®) inside the corporation.

Using the same protocols inside and outside of the corporate network leads to a seamless computing environment that scales to tens of millions of users and promotes competition among vendors.

Perhaps the most significant untapped resource on the Internet is the human one. There is often a need to access either a person or an application monitored by a person. Without a directory architecture that enables users to find each other or their applications easily, electronic communication cannot live up to its full potential. The virtues of a truly open directory architecture are also sorely missed inside corporations where user information is fragmented in pockets of proprietary technology.   On the one hand, MIS managers must struggle with maintaining employee information across a myriad of incompatible systems. On the other, not only must MIS managers maintain their own employee profiles and access authority, they are often responsible for providing the capability for users outside their organizations to access specific information and applications. Directories that can be maintained by each individual entity so they are always current would alleviate some of the problem. Today, the primary challenge is to provide a process to allow directories to talk to each other and help find the needed person or application.

The benefits to be gained from electronic directories are:

- Maintenance and administrative costs in managing outside users will go away and save the costs involved.

- Corporate users will be able to communicate faster with customers, suppliers, and trading partners.

- Directories will be more accurate and alleviate the problem of data going to the wrong person or application.

The goals of a directory strategy are to provide and/or enable:

- Logically centralized user management; that is, one place to add, move, change, or delete users.

- Complete interoperability, including client-to-server communication and replication.

- Internet scalability and interoperability.

- No vendor lock-in; in other words, no proprietary ownership or control of the directory protocols.

- Free reference implementations of available directory protocols in order to maximize competition.

- Minimum time to set up new users and give them properly controlled access to your information.

There are two major pieces to a directory services strategy. The first is to determine the information that needs to be housed in a directory, and the second is to determine how to get to that information.

The Automotive Industry Guideline for Directory Services will provide a common way to obtain business information accessed through a web browser. The implementation of this guideline assumes that a repository of electronic commerce application information may be valuable to both manufacturers and suppliers. Some of the application items the repository might contain are:

- Versions

- Releases

- Status

- Policies and business rules

- Implementation guidelines and version controls

- Buyer's guides

Other information that may be retrieved:

- Key company business contacts

- Key company technical contacts

- Company's name

- Company's line of business

- Phone numbers

- Fax numbers

- Company address

- Specific locations and characteristics of the locations (e.g., dock height and identification, directions to location, times of operation, etc.)

- Digital Certificates

The availability of the Directory Services will allow individuals to use their keyboards to do the task of locating and contributing to information pertinent to the automotive industry. A company using the common directory formats and technologies to call a directory will be able to retrieve and gather key business information to assist in business-to-business application planning and development with their key trading partners. Directory Services will also offer companies the ability to administer their applications that reside in their directory and link to applications in other directories. The directory services implementation will have a common interface/structure for companies to maintain their internal directories. This will result in greater accuracy, less maintenance, and reduced required systems capacity.

Just as a telephone company's paper version of the Yellow Pages offers information in an easy-to-follow format, linking company directories in a common format that can be accessed with a web browser will allow for valuable information to be shared electronically by everyone. Automotive companies will find maximum benefit when using Directory Services for their business needs.

3.0   INITIAL AUTOMOTIVE DIRECTORY OBJECTS

The initial directory structures for the automotive industry are built around three key entities: persons, companies, and locations. A main requirement is for standardized object classes and attributes to represent persons, companies, and locations in LDAP-compatible directories so that this information can be shared. The requirements and schema for each entity vary.

3.1   Person Object Type

A *person* is an entity that a user can do business with such as creating contracts, passing data, collaborating, or addressing any number of other business process needs. The AIAG has adopted the `inetOrgPerson` object class as the basis to describe persons related to automotive business. `Persons` will be associated with `Companies,` `Locations,` `Applications,` and other objects.

The *Person* is the most basic object in the directory. It is the digital representation of an actual human being, and contains information intended to facilitate any number of business processes (creating contacts, exchanging information, collaborating, and so forth). This object class will be maintained by routines that access a company's directory services. The routines will typically allow adding/modifying/deleting a person and information about that person.

In order to facilitate inter-directory communication (potentially including direct referrals between directories) it is advantageous to define a certain minimum industry-common schema for the *Person* objects, along with a standard approach for each company to extend the schema as required by internal applications.

The schema for person objects is laid out in Appendix A. The discussion provides a detailed perspective on how to extend the schema. In addition, Distinguished Name structure best practices are examined.

3.2   Company Object Type

A `Company` is a kind of "virtual person", an entity that one may do business with: enter into contracts with, complete transactions with, etc. `Companies` will be associated with `Persons,` `Locations,` `Applications,` and other objects.

The `Company` object is intended to provide a representation of this business entry in a Directory other than that company's own directory.

For example, if Company *[Acme]* has a directory (with a base Distinguished Name of `o=Acme.com`) and Company *[Acme]* does business with Company *[Bar]* (or keeps representations of `People` working for Company *[Bar]* in its directory, or representations of any other objects associated with Company *[Bar]* in its directory), then there should exist an entry in Company *[Acme's]* directory that represents Company *[Bar]*. In this manner, the relationship between directory objects and Company *[Bar]* can be

maintained with an attribute of that object that contains the DN for the Company [Bar] Company object entry.

Changing relationships between other objects and Companies thus becomes a simple matter of changing the associated attributes in the target object's entry to contain the Domain Name of the correct Company. Applications that manage the relationships of objects to Companies do not need to be responsible for managing the attributes of the Company object itself.

Company objects should be kept as atomic as possible. As Companies are "virtual people" and have no real existence outside the legal systems of the world, they tend to change names, consolidate, fragment, and vanish with a much higher frequency than real physical objects such as people or buildings. Accordingly, no effort should be made to represent a given company's internal organizational structure within the scope of the Company object with any higher degree of granularity than simple parent-child relationships.

For example, there may exist a Company object representing "General Motors" and a separate Company object representing "Pontiac" (a separate child Company of General Motors), but there should not be a separate Company object representing "The Pontiac Human Resources Department" unless that department exists as a stand-alone business entity in its own right.

The schema for company objects is laid out in Appendix B.

3.3   Location Object Type

A location object is that place within the business community where business takes place, whether it be a corporate office, a manufacturing plant, or a loading dock. The Location object type is a physical area or object that People and Companies (and indeed, other objects) may be associated with. Unlike the l objectclass attribute currently defined by X.521, Location objects are intended to represent "localities" with a level of granularity of the order of real property and a limited level of subdivisions of real property, rather than of the order of cites, counties, etc.

Location objects are intended to describe buildings, structures, and areas that are "owned", used, or are in some manner subject to interest by the company maintaining the directory that contains their entries.

Note that unlike Person objects, which may be classed and subclassed in many various ways but ultimately remain a single physical object, Locations contain a number of distinctly different physical representations - a building is not a ball field. Furthermore, some Locations act as containers for other Locations; although there may be no parent-child relationship on the objectclass level between the parent and the child, there is a relationship between the container and the contained that must be maintained within the object itself.

The schema for Location objects is laid out in Appendix C.

4.0   GUIDELINE CHANGE PROCESS

This document is a living document. The schema, object classes, and attributes defined here will change over time, as business needs change. Appendix D defines the process through which AIAG will handle these changes. The model is similar to that used to make changes to the AIAG Guidelines for EDI transaction sets.

APPENDIX A. PERSON OBJECT TYPE DESCRIPTIONS

A.1   Person Object Class Description

```
objectclass top
      oid 2.5.6.0
      requires
            objectClass
      allows
            aci

objectclass person
      oid 2.5.6.6
      superior top
      requires
            sn,
            cn
      allows
            description,
            seeAlso,
            telephoneNumber,
            userPassword

 objectclass organizationalPerson

      oid 2.5.6.7
      superior person
      allows
            destinationIndicator,
            facsimileTelephoneNumber,
            internationalISDNNumber,
            1,
            ou,
            physicalDeliveryOfficeName,
            postOfficeBox,
            postalAddress,
            postalCode,
            preferredDeliveryMethod,
            registeredAddress,
            st,
            street,
            teletexTerminalIdentifier,
            telexNumber,
            title,
            x121Address

objectclass inetOrgPerson
      oid 2.16.840.1.113730.3.2.2
      superior organizationalPerson
      allows
            audio,
            businessCategory,
            carLicense,
            departmentNumber,
```

```
        employeeType,
        employeeNumber,
        givenName,
        homePhone,
        homePostalAddress,
        initials,
        jpegPhoto,
        labeledURI,
        manager,
        mobile,
        pager,
        photo,
        preferredLanguage,
        mail,
        roomNumber,
        secretary,
        uid,
        x500uniqueIdentifier,
        userCertificate
        userCertificate;binary,
            usersMimeCertificate;binary

objectclass  AIAGperson
  parent InetOrgPerson
  allows
    name
     AIAGCompanyRepresenting
    officeAddress
    officeTelephoneNumber
    officeFacsimileNumber
    homeTelephoneNumber (optional)
    pagerNumber
    mobilePhoneNumber
    title
     AIAGApplicationResponsibility
    jpegPhoto (if required)
    description (as necessary)
```

A.2   Recommended Distinguished Name Format

Classic X.500 naming practice typically defines the Distinguished Name (DN) structure for a person as:

*[some id]=[some unique value],**ou**=[some organizational unit],**o**=[organization]*

Example:

```
uid=joe.user,ou=marketing,o=acme.com
```

However, in practice, this approach suffers from several flaws. First, nothing in the DN identifies this object as a *Person.* Second (and most importantly) the DN now mirrors both the organizational structure of the company **and** the object's position within the structure tree. Either one of these (the structure, or

the position within the structure) may change during the lifetime of the object. Such a change would require that the DN of the object change, which further requires every reference to that object (which may be stored in directories external to the main one) to be changed as well. This can have significant impact on applications that rely on a particular naming strategy, such as a Public Key Infrastructure.

There is great value to having the DN of an object remain constant throughout its lifespan. Given this perspective, the recommended DN structure for all objects in Directories is:

*[some identifier]=[some unique value],**ou**=[objecttype],**o**=[directory domain]*

In the specific case of *Persons,* that translates to:

*[some identifier]=[some unique value],**ou=people,o**=[directory domain]*

Example:

```
daimlerchryslerDGID=0000000002,ou=people,o=dcx.com
```

The attribute used to form the "[some identifier]" portion of the DN is left to the discretion of the implementers of the directory - as long as its value is unique across the entire "people" portion of the namespace. "`cn`" is thus a poor choice, as there will be many name collisions across the people space in any directory of reasonable size. The `daimlerchryslerDGID` in the example is a sequential, numerical counter incremented each time a *Person* is added to the Directory, and this has worked well. Similar strategies may be employed by other implementations ; as long as the value is unique across the entire *people* space, and unlikely to change value during the object's lifetime, the actual attribute used is of no consequence.

A.3   Schema Definition

The base objectclass adopted by the AIAG is `inetOrgPerson`. This objectclass defines a large number of common attributes for People, and these attributes are the de facto worldwide standard for directory-enabled applications.

`inetOrgPerson` is a child object of the `organizationalPerson` object class. `organizationalPerson` is in turn a child of `person`, which is a child of `top`. The `person` object class requires the attributes `cn` (common name) and `sn` (surname) be populated.Thus the minimum attribute set for a *Person* object becomes:

*dn: [some identifier]=[some unique value],**ou=people,o**=[domain]*
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
ou: people
cn: [usual first name] [surname]

sn: [surname]

Example:

```
dn: daimlerchryslerDGID=0000000002,ou=people,o=dcx.com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
ou: people
cn: Dennis Grant
sn: Grant
```

A.4   Schema Extension and Naming Conventions

The minimal schema defined in A.3 is almost certainly insufficient to describe the *Person* so modeled with the required level of detail. Each company will need to populate additional attributes specific to that company's data modeling requirements. This will require company-specific extensions to the *Person* schema.

The first layer of customization is to be accomplished via the *fooPerson* objectclass, a child of `inetOrgPerson`, where "foo" is the name of the company doing the extension, spelled out in full. The *fooPerson* object class contains all company-defined attributes that are common to all *People* stored in Foo's directory.

Example:

```
daimlerchryslerPerson
fordPerson
```

A "*fooPerson*" is not "a person who works for company Foo" but rather "a person stored in company Foo's directory, with Foo-defined attributes."

If further specialization is required, using additional attributes not common to the entire People population (often mutally exclusive, but not always) further auxiliary object classes (children of *fooPerson)* may be added, each with their own attribute sets.

The value of auxiliary object classes may thus be used to distinguish between different kinds of *People*.

Every attempt must be made to populate the standard attributes defined by `inetOrgPerson` with their intended values rather than re-defining them with local names. For example, an "email address" should be populated in the standard attribute `mail` rather than the locally-defined `foocorpEmailAddress`.

All attributes of any locally defined object classes are also to be prefixed with the company name spelled out in full. Furthermore, all acronyms or abbreviations used in locally defined attribute names are also to be spelled out in full.

Example:

Company Foo defines its unique identifier as `fooDGID` which is common to all People in its directory. It has employees, who have a pay rate code, and customers, who have a date of last product purchase.

Thus, company Foo defines an objectclass for employees, `fooEmployeePerson,` which allows the attribute `fooPayRateCode`. It also defines an objectclass, `fooCustomerPerson,` which allows the attribute `fooLastPurchaseDate`.

Examples of these objects would look like:

```
dn: fooDGID=0000000005,ou=people,o=foo.com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
objectclass: fooPerson
objectclass: fooEmployeePerson
ou: people
cn: Marcus Merideth
sn: Merideth
fooDGID=0000000005
fooPayRateCode: a5

dn: fooDGID=0000000010,ou=people,o=foo.com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
objectclass: fooPerson
objectclass: fooCustomerPerson
ou: people
cn: Mary Bigglesworth-Smith
sn: Bigglesworth-Smith
fooDGID=0000000010
fooCustomerLastPurchaseDate: 20020212
```

Note that in this case, the two auxiliary object classes are not mutually exclusive. If the employee "Joe Bloggins" were to purchase something, he would have the objectclass `fooCustomerPerson` added to his entry, and the attributes associated with `fooCustomerPerson` would now become available for writing and reading.

For mutually exclusive objects (for instance, a `fooRetiredEmployeePerson`) the exclusivity of the objectclasses and their attributes must be enforced by the directory authoring systems; the directory server itself will not necessarily do it for you.

This extension model may be applied to any object in the directory, not just people, and it is in fact a core feature of this standard.

APPENDIX B. COMPANY OBJECT TYPE DESCRIPTIONS

B.1   Company Object Class Definition

```
objectclass Company
     parent Top
     must contain
          cn
          objectclass
     may contain
          companyUniqueIDCode        ces
          companyDUNSNumber          ces
          companyEANNumber           ces
          companyStockTickerCode     ces
          companyStockExchange       cis
          companyImmediateParent     dn
          companyHomeCountry         cis
          companyIsActive            cis
          companyPrimaryContact      dn
          companyPublicDirectoryURL  ces
          companyPublicHTTPURL       ces
          companyIndustryCategory    cis
          description
          companyDomainName          ces
```

B.2   Company Attribute Descriptions

CompanyUniqueID: An identification code or number that uniquely identifies the `Company` within the scope of the directory. Not required to be unique across the set of all `Companies` in all directories.

CompanyDUNSNumber: The Dun and Bradstreet number (if any) associated with the `Company`.

CompanyEANNumber: The EAN number (if any) associated with the `Company`.

CompanyStockTickerCode: The letter code (if any) under which the `Company` is traded on a public stock exchange.

CompanyStockExchange: The name of the stock exchange (if any) on which the `Company` is traded.

CompanyImmediateParent: The DN of the `Company` (if any), which serves as the immediate parent of the entry.

CompanyHomeCountry: The ISO country code of the country in which the `Company` is registered/incorporated.

CompanyIsActive: Boolean flag (Yes textbar No) indicating if the entry is to be considered active.

CompanyPrimaryContact: DN to object (usually a `Person`, but not limited as such) that serves as a primary contact for the `Company` within the scope of the directory.

CompanyPublicDirectoryURL: LDAP URL to the publicly exposed (if any) LDAP directory for this `Company`. Should include searchbase.

CompanyPublicHTTPURL: Http URL to the `Company's` public web site (if any).

CompanyIndustryCategory: Text description of the industry category(ies) the `Company` falls into companyDomainName The identifier used in the `Company's` own o attribute in its own directory.

B.3   Recommended Distinguished Name Format

It is recommended that `Company` DNs be kept in a separate branch of the tree based on object type. It is also recommended that whatever attribute is used as an identifier in the DN not be subject to changes outside of the directory's control. For this reason, "company name" does not make a good candidate for inclusion in the DN, as companies tend to change names fairly frequently and often without warning. A far better candidate is `companyUniqueID` because it is under the control of the company maintaining the directory and often was assigned by other business systems well before the company directory was established. It thus provides a direct hook for directory entries into legacy, non-directory-aware systems.

For a directory maintained by *[Arius]* (with a base DN of `o=arius.com`), the entry for a company with a unique ID `Acme` assigned to it by *[Arius]*, the DN would therefore be:

```
companyUniqueID=Acme,ou=companies,o=arius.com
```

APPENDIX C. LOCATION OBJECT TYPE

C.1 Location Object Class Descriptions

The following objectclasses are defined as `Locations`:

Location: the parent objectclass

Complex: a container objectclass containing at least two adjacent Sites

Site: a location with a legal definition and a singular postal address. Each Site must be subclassed as at least one of the following:

- Land
- FunctionalArea
- Structure
- Building

> **Land:** an undeveloped portion of the earth's surface with clearly defined legal boundaries.
>
> **FunctionalArea:** a developed area of land with an intended use that typically does not contain `Buildings` or `Structures`. Each `functionalArea` must be subclassed as at least one of:
>
> - **SportsField:** a `functionalArea` where a sport is played. Must be further identified by which sport it is used for (baseball, basketball, etc.).
> - **ConfinedSpace:** a `functionalArea` used to manage waste or hazardous materials.
> - **DetentionPond:** a `functionalArea` used to displace rainwater.
> - **Lake:** a `functionalArea` used to retain water and aquatic life.
> - **LandingArea:** a `functionalArea` suitable for the landing of VTOL aircraft such as helicopters and balloons.
> - **LandingStrip:** a `functionalArea` that can accommodate the landing of non-VTOL aircraft such as planes and gliders.
> - **Landscape:** a `functionalArea` containing decorative or protective earth works and/or flora (such as a garden).
> - **Landfill:** a `functionalArea` used as a disposal area for waste material, usually buried.
> - **ParkingLot:** a `functionalArea` used to store vehicles.
> - **RailroadTrack:** a `functionalArea` used to convey railroad vehicles.
> - **RetentionPond:** a `functionalArea` used to retain rainwater.
> - **Road:** a `functionalArea` used to convey vehicular traffic.

- **StorageLot:** a `functionalArea` used to store material.

- **TestTrack:** a `functionalArea` used as a test or competition site for vehicles.

- **Waterway:** a `functionalArea` used to convey water-borne traffic.

- **Woods:** a `functionalArea` that is set aside for trees, brush, and wildlife.

- **Crops:** a `functionalArea` used to grow food crops.

- **Pasture:** a `functionalArea` used to graze livestock.

- **DefenseWorks:** a `functionalArea` used as a protective area, such as a minefield or a trench works.

`Structure`: An artificial construct that is not a building. Each `Structure` must be subclassed as at least one of the following:

- **Art:** a decorative or illustrative object or monument.

- **ContainmentTank:** a `structure` used to contain liquid, granular, or gaseous material.

- **CommunicationsTower:** a `Structure` used to transmit electro-magnetic signals such as radio waves.

- **Fence:** a `Structure` used to prevent intrusion or escape and/or as a boundary marker.

- **Gate:** a `Structure` used to allow limited access into another area.

- **Lighting:** a `Structure` used to illuminate an area.

- **Obstacle:** a `Structure` used to impede or prevent passage.

- **ObservationTower:** an elevated `Structure` used to provide an observation point.

- **Signage:** a `Structure` used to display information or advertising.

- **TankFarm:** a `Structure` housing multiple containment tanks.

- **TransmissionTower:** a `Structure` used to elevate power lines and/or contain electrical transformers.

- **WaterTower:** an elevated `Structure` used to contain water.

`Building`: a normally roofed and walled architectural structure intended to house people and/or material. A `Building` may act as a container object for one or more of the following:

- **Section:** an enclosed area within a `Building` that may contain further sub-objects.

- **Bay:** an area within a `Building` intended to provide vehicular access to the building's interior.

- **Hanger:** an area within a `Building` intended to house vehicles.

- **Node:** an area within a `Building` that serves as an access point to one or more Sections.

- **Room:** the smallest level of subsection within a `Building`.

- **Floor:** an enclosed area within a `Building` separated in the vertical plane.

C.2 Objectclass Descriptions

```
objectclass location
     parent top
     requires
       cn
     allows
       locationBoundaries                        cis
       locationMapIdentifier                     ces
       locationOwner                             dn
       locationUniqueID ces
       description


 objectclass complex
     parent location
     requires
       complexMemberSites
     objectclass site
        parent location
        requires
       l
       street
     allows
       postalCode
       st
       c


 objectclass land
     parent site


 objectclass functionalArea
     parent site


objectclass sportsField
     parent functionalArea
     requires
       sportsFieldType                           cis


objectclass confinedSpace
     parent functionalArea


objectclass detentionPond
     parent functionalArea


objectclass lake
     parent functionalArea
     allows
       lakeIsNavigable
       lakeMaxDepth

objectclass landingArea
     parent functionalArea
     allows
       landingStripSurfaceType                   cis
```

```
        landingStripIsControlledAirport      cis
        landingStripElevation                cis
        landingStripAirportID                css
        landingStripMagneticVariation        cis
        landingStripUnicomFrequency          cis
        landingAreaFacilities                cis
        landingAreaPadDimensions             cis


objectclass landingStrip
     parent functionalArea
     requires
       landingStripRunwayNumbers cis
     allows
       landingStripSurfaceType             cis
       landingStripAirportID               cis
       landingStripElevation               cis
       landingStripIsControlledAirport     cis
       landingStripUnicomFrequency         cis
       landingStripMagneticVariation       cis
       landingStripRunwayWidth             cis
       landingStripRunwayLength            cis


 objectclass parkingLot

     parent functionalArea
     allows
        parkingLotVehicleCapacity          cis
        parkingLotIsControlledAccess       cis


 objectclass landfill
     parent functionalArea
     allows
        landfillContainsHazardousWastes


 objectclass retentionPond
     parent functionalArea


 objectclass railroadTrack
     parent functionalArea
objectclass road
     parent functionalArea


 objectclass storageLot
     parent functionalArea


 object testTrack
     parent functionalArea


objectclass waterway
     parent functionalArea
     allows
       waterwayDepth                         cis
       waterwayWidth                         cis
       waterwayCurrentSpeed                  cis
```

```
objectclass crops
      parent functionalArea

objectclass woods
      parent functionalArea

objectclass defenseWorks
      parent functionalArea

objectclass pasture
      parent functionalArea

objectclass landscape
      parent functionalArea

objectclass structure
      parent site

objectclass art
      parent structure

objectclass containmentTank
      parent structure

objectclass communicationsTower
      parent structure

objectclass fence
      parent structure
      allows
         fenceConstructionType                cis
         fenceIsElectrified                   cis

objectclass gate
      parent structure
      allows
         gateIsManned                         cis

objectclass lighting
      parent structure

objectclass observationTower
      parent structure
      allows
         observationTowerHeight               cis

objectclass obstacle
      parent structure

objectclass signage
      parent structure
      allows
         signageIsLighted                     cis
```

```
objectclass transmissionTower
      parent structure
      allows
         transmissionTowerHasTransformer      cis
         transmissionTowerLinesCount          cis

objectclass waterTower
      parent structure
      allows
         waterTowerFluidCapacity              cis

objectclass tankFarm
      parent structure
      allows
         tankFarmAssociatedTanks              dn

objectclass building
      parent site
      allows
         buildingNumberOfFloors               cis

objectclass buildingSubElement
      parent location
      requires
         buildingSubElementAssociatedBuilding dn

objectclass section
      parent buildingSubElement

objectclass bay
      parent buildingSubElement

objectclass hanger
      parent buildingSubElement
      allows
         hangerVehicleCapacity                cis

objectclass node
      parent buildingSubElement
      allows
         nodeAttachedBuildingSubElements      dn

objectclass room
      parent buildingSubElement
      allows
         roomAssociatedFloor                  cis

 objectclass floor
      parent buildingSubElement
      allows
         floorNumber                          cis
```

C.3   Location Attribute Descriptions

locationBoundaries: a series of coordinate pairs that describe in some manner the boundaries of the `Location`.

locationMapIdentifier: the unique identifier for the map used to determine the coordinates used in the `locationBoundaries` attribute.

locationUniqueID: a code that uniquely identifies the `Location` within the scope of the containing directory.

locationOwner: the DN of a company or organization that "owns" or is associated with the `Location`.

complexMemberSites: a multi-valued list of the DNs of all the sites that comprise the `Complex`.

sportsFieldType: the sport or sports that take place on this field.

lakeIsNavigable: a flag indicating if the `Lake` is connected to other waterways accessible by waterborne traffic.

lakeMaxDepth: the depth of water at the deepest point.

landingStripSurfaceType: paved, concrete, grass, etc.

landingStripIsControlledAirport: a flag indicating if the `landingArea` or `landingStrip` controls the airspace around it.

landingStripElevation: the elevation above sea level.

landingStripAirportID: the universal assigned identifier for the associated `Airport`.

landingStripMagneticVariation: the difference between magnetic north and true north at this `Location`.

landingStripUnicomFrequency: the radio frequency on which this `Location` broadcasts its control information.

landingAreaFacilities: a description of the service facilities at this `Location`.

landingAreaPadDimentions: the size of the actual landing surface.

landingStripRunwayNumbers: the identification numbers for the runway(s) at this `Location.`

landingStripRunwayWidth: the width of the primary runway.

landingStripRunwayLength: the length of the primary runway.

landfillContainsHazardousWastes: a flag indicating the presence of hazardous material.

tankFarmAssociatedTanks: a list of DNs of containmentTanks contained in the `tankFarm`.

parkingLotVehicleCapacity: the number of vehicles that can be stored on this lot.

parkingLotIsControlledAccess: a flag indicating if there exists a gate or other device limiting access to the lot.

waterwayWidth: the width of the waterway at its narrowest point.

waterwayDepth: the depth of the waterway at its shallowest point.

waterwayCurrentSpeed: the speed of the current (if any) in the waterway.

fenceConstructionType: chain link, concrete, wood, etc.

fenceIsElectrified: a flag indicating if the `Fence` is electrified.

gateIsManned: a flag indicating if the `Gate` is normally manned by a human guard.

signageIsLighted: a flag indicating if the `Signage` is illuminated at night.

transmissionTowerHasTransformer: a flag indicating if the `transmissionTower` mounts a transformer.

transmissionTowerLinesCount: the number of transmission lines the `Tower` supports.

waterTowerFluidCapacity: the amount of fluid the `Tower` can contain.

buildingNumberOfFloors: the number of `Floors` in the building.

buildingSubElementAssociatedBuilding: the DN of the `Building` containing this `buildingSubElement.`

hangerVehicleCapacity: the number of vehicles the `Hanger` can house.

nodeAttachedBuildingSubElements: the DNs of the `buildingSubElements` connected by the node.

floorNumber: the number of the `Floor`.

C.4   Recommended Distinguished Name Format

It is recommended that `Location` DNs be kept in a separate branch of the tree based on object type. It is also recommended that whatever attribute is used as an identifier in the DN not be subject to changes outside of the directory's control. Good candidates are `cn` and `locationUniqueID`.

For a directory maintained by *[Arius]* (with a base DN of `o=arius.com`), the entry for a `Building` with a unique ID *[Acme]* assigned to it by *[Arius]*, the DN would therefore be:

```
locationUniqueID=Acme,ou=locations,o=arius.com
```

Example Entries

For the DaimlerChrysler Center Line ITM building, as it might be stored in the Arius, Inc. directory:

```
dn: cn=ITM Center Line,ou=locations,o=arius.com
objectclass: top
objectclass: location
objectclass: site
objectclass: building
cn: ITM Center Line
ou: locations
street: 25999 Lawrence Ave
l: Center Line
st: Michigan
c: USA
postalCode: 48015-0303
buildingNumberOfFloors: 1
locationOwner:
companyUniqueID=DaimlerChrysler,ou=companies,o=arius.com
```

For the helipad at the Detroit Medical Center:

```
dn: cn=Detroit Medical Center Helipad,ou=locations,o=arius.com
objectclass: top
objectclass: location
objectclass: site
objectclass: functionalArea
objectclass: landingArea
cn: Detroit Medical Center Helipad
ou: locations
locationBoundaries: 42deg12minN, 83deg20minW
landingStripSurfaceType: concrete
landingStripIsControlledAirport: no
landingStripElevation: 630 ft
landingStripAirportID: 5MI0A
landingStripMagneticVariation: 06W
landingAreaPadDimensions: 80 ft X 80 ft
l: Detroit
st: Michigan
street: 420 St. Antoine
c: USA
```

APPENDIX D. GUIDELINE CHANGE PROCESS

In order to accommodate the needs of the trading partners in the automotive industry, the AIAG will use a Guideline Change Process to make sure the Directory Services Guideline meets the needs of its users.

If any trading partner deems a change necessary that partner will submit a Maintenance Request to AIAG (see attached Change Request below). The Program Manager or Executive Loan responsible for the Directory Services document will forward the request to the appropriate Project Team or Steering Committee for action.

The submitter of the Maintenance Request will document the change requested and the business reason for the change including any financial impacts the change may incur. Since the guideline is meant for company-to-company communication, only those changes that deal with intercompany directory interfaces will be addressed. The Project Team or Steering Committee will determine the validity and/or applicability of the request and do one of the following:

- Forward it to the appropriate Work Group for review.

- Send it back to the submitter for further information or clarification.

- Reject it including the reason(s) for rejection.

If the request is forwarded, the appropriate Work Group will address the request for change as the normal course of business. If there is no existing Work Group, the Project Team or Steering Committee will take action to initiate a call to appropriate personnel inside their companies to create an ad hoc committee to review the request for validity and inclusion in the Guideline.

ABOUT AIAG

Purpose Statement

To provide an open forum where members cooperate in developing and promoting solutions that enhance the prosperity of the automotive industry. Our focus is to continuously improve business process and practices involving trading partners throughout the supply chain.

Core Values

- **People –** Our strength comes from passionate and personally committed volunteers and staff. We provide an environment of integrity, trust, teamwork and mutual respect to foster open, frank communication as we achieve consensus on industry needs and solutions.

- **Innovation –** With a sense of urgency, we drive and support the development and implementation of common, leading-edge solutions that provide value to the automotive industry and its customers.

- **Excellence –** We provide quality and excellence in all we do and how we do it.

We do what's right for the industry!

AIAG Organization

AIAG is made up of a board of directors, an executive director, associate directors, a full-time staff, and volunteers serving on project teams. Under the direction of the executive director, associate directors, along with the managing director, department managers and program managers, plan, direct, and coordinate the association's activities. The executive director and loaned executives are *on loan* from member companies for varied lengths of time.

AIAG Projects

Volunteer Committees focus on business processes or supporting technologies and methodologies. They conduct research, develop, publish and provide training on standards, conventions, standard business practices, white papers and guidelines in the areas of automatic identification, CAD/CAM, EDI/electronic commerce, continuous quality improvement, materials and project management, returnable containers and packaging systems, and transportation/customs.

AIAG - An Association Fostering Total Supply Chain Partnering

Automotive Industry Action Group
26200 Lahser Road, Suite 200
Southfield, MI 48034
Phone: (248) 358-3570 • Orders: (248) 358-3003 • Fax: (248) 358-3253

MAINTENANCE REQUEST

Note: All AIAG publications are reviewed periodically. Complete form and return it to the Volunteer Programs Department Manager for consideration.

Name of Submitter: _____ Date: _____

Company: _____

Company Address: _____

Phone: _____ Fax: _____ E-mail: _____

════════════ **MAINTENANCE REQUEST** (Use additional sheets if necessary) ════════════

Page Number of Change: _____

Document Currently Reads: _____

_____

_____

Recommended Changes/Should Read: _____

_____

_____

Recommended Additions: _____

_____

_____

Reason for Change: _____

_____

_____

Signature of Submitter: _____

══════ **DISPOSTION** (AIAG use only) – Completed forms are filed in the final document maintenance file. ══════

Program Owner Assigned to: _____ Log #: _____

Program Owner's Recommendation: _____

_____

Final Disposition: _____

_____

Comments: _____

_____

**Automotive Industry Action Group • 26200 Lahser Road • Suite 200 • Southfield, MI 48034
Telephone: (248) 358-3570 • Fax: (248) 358-3253
www.aiag.org**

VP-MR   **Revision: 04 Dated: 05/11/01**