The University of Kansas

Information and
Telecommunication
Technology Center

Technical Report

ACE Service Directory Interface Specification Version 1.0

Renzo Hayashi, Leon Searl, and Gary Minden

ITTC-FY2001-TR-23150-06

May 2001

Project Sponsor:
U.S. Air Force and the Defense Advanced Research
Projects Agency under contract no. F30602-00-2-0581
and The National Science Foundation
under grant EIA-9972843

Table of Contents

1 Introduction

This document shall formally specify the interface for the ASD (ACE Service Directory) Daemon that serves as a directory listing of services available to other ACE daemons and clients. ACE daemons may register their services within the ASD and other daemons or clients wishing to use ACE services may look them up in the ASD as well.

2 Specifications

This section describes the interface that other daemons shall use to interact with the ASD and exchange ACE service information with it.

2.1 Problem Domain

As soon as an ACE daemon starts up and wishes to be available to the ACE environment, it must register its existence and attributes with some kind of directory service so that it can be made available to other daemons. Also, if it wishes to utilize any other service or communicate with any other device or daemon it must know where to look for it. That is the purpose of the ASD – a centralized location where any and all services may exchange knowledge about their existence and capabilities.

2.2 Result Arguments Semantics

This section is intended to outline the semantics of the arguments supplied in a result command. This command sent back to detail the results of a previously issued command. The arguments in the result command are a common reply to many ACE commands sent by clients and therefore shall be referred to by other sections in this document.

S-2.2.1 Results Arguments

'status' – Specifies the status, success or fail of the previously issued command.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_WORD
Allowed Numbers Range	: N/A
Allowed Enum Values	: { success, fail }
Mutually Dependent	: None
Mutually Exclusive	: None

'msg' – A message about the error that has occurred. It is more specific that an error number.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { cmdErrorNo }
Mutually Exclusive	: None

'cmdErrorNo' – An error number for a failure of the command.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_INT
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { msg }
Mutually Exclusive	: None

2.3 Query Interface Specifications

This section describes the commands that the ACE Service daemons shall use to inquire about the availability of specific ACE services along with their capabilities and attributes.

S-2.3.1 ServiceLookup command

Command Name: 'ServiceLookup'

Description: This is a command used by an ACE Service Daemon to look for another service within the ACE Service Directory.

Arguments:

'name' - The name of the service that is being sought.

Occurrences	: {0, 1} (Optional)
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'classHierarchy' - The entire class hierarchy of the ACE service that is being sought. Each element in the array contains the name of a level of the ACE Service Class hierarchy. The first element is the root class level. The last/leaf element is the class level of the specific service.

Occurrences	: {0, 1} (Optional)
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'location' – The physical location of the service (if any).

Occurrences	: {0, 1} (Optional)
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

Example: ServiceLookup name=”Camera1” location=”Reading Room”;

This shall return all services located in the “Reading Room” that are called “Camera1”.

This lookup command may be issued with one, all, or none or the above arguments. If any of the above arguments are not supplied, then the search shall not be performed on those arguments. For instance, if only the argument ‘name’ is supplied, then all services within the ASD that match that name will be returned, irrespective of their class hierarchy or location. If no arguments are supplied, all services currently within the ASD shall be returned. The more arguments supplied the smaller the result recall.

S-2.3.2 ServiceLookupResult command

Response Command Name: 'ServiceLookupResult'

Description: This is the return command resulting from a ServiceLookup command. It returns the query results for all matching services and their corresponding information.

Arguments:

'name' - The names of the services that matched the query.

Occurrences	: CMDARG_OCCURS_OPTIONAL
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'classHierarchy' - The class hierarchy of the services that matched the query. Each element is an array of strings of the same format as the 'classHierarchy' query argument in S-2.3.1 above.

Occurrences	: CMDARG_OCCURS_OPTIONAL

Data Type	: CMDVAL_STRING_VECTOR
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'location' - The location names where the services that matched the query are located.

Occurrences	: CMDARG_OCCURS_OPTIONAL
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'address' - The ACE network address that clients must use to connect to the service.

Occurrences	: CMDARG_OCCURS_OPTIONAL
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

Further arguments include the **results arguments** detailed in section 2.2.

For the ‘name’, ‘classHierarchy’, ‘location’, and ‘address’ arguments the nth element of each array is information for a single service. Other than the correspondence of elements between the arrays there is no order for the array elements.

2.4 Mapping Interface Specification

2.4.1 Service Daemon to ACE Service Directory Commands

This section contains commands that ACE Services daemons can send to the ACE Service Directory Daemon. These commands are related to ASD management and how services interface with the ASD to register themselves and make themselves known to the ACE environment.

S-2.4.1.1 ServiceRegister command

Command Name: 'ServiceRegister'

Description: This is a command for an ACE Service Daemon to register itself with the ACE Service Directory so that clients may later query the ACE Service Directory to find out about the service.

Arguments:

'name' - The name of the service that is registering.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { address, classHierarchy, location }
Mutually Exclusive	: None

'address' - The ACE network address of the registering service.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { name, classHierarchy, location }
Mutually Exclusive	: None

'classHierarchy' - The ACE Service Class Hierarchy of the service being registered. This argument is an array type. Each element in the array contains the name of a level of the ACE Service Class hierarchy. The first element in the array is the hierarchy root level. The last/leaf element is the class of the service.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { name, addresss, location }
Mutually Exclusive	: None

'location' – The physical location (if any) where the service is located.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { name, addresss, classHierarchy }
Mutually Exclusive	: None

S-2.4.1.2 ServiceRegisterResult command

Response Command Name: 'ServiceRegisterResult'

Description: This is the return command resulting from a ServiceRegister command. It simply returns the status of the issued ServiceRegister command along with pertinent lease information.

Arguments:

'leaseTime' – The duration of the service lease. That is, for how long the service information shall be kept in the ASD before it is automatically removed by the ASD daemon. This is in milliseconds.

Occurrences	: CMDVAL_OCCURS_OPTIONAL
Data Type	: CMDVAL_INT
Allowed Numbers Range	: { 5000, 3,600,000 }
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

Arguments also include the **results arguments** detailed in section 2.2.

S-2.4.1.3 ServiceUnregister command

Command Name: 'ServiceUnregister'

Description: This is a command for an ACE Service Daemon to unregister itself from the ACE Service Directory once it begins to remove itself from the ACE environment (usually done on daemon shutdown).

Arguments:

'name' - The name of the service that is unregistering.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { address, classHierarchy, location }
Mutually Exclusive	: None

'address' - The ACE network address of the service being unregistered.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { name, classHierarchy, location }
Mutually Exclusive	: None

'classHierarchy' - The ACE Service Class Hierarchy of the service being unregistered. This argument is an array type. Each element in the array contains the name of a level of the ACE Service Class hierarchy. The first element in the array is the hierarchy root level. The last/leaf element is the class of the service.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { name, addresss, location }
Mutually Exclusive	: None

'location' – The physical location (if any) where the service is located.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: { name, addresss, classHierarchy }
Mutually Exclusive	: None

S-2.5.1.4 ServiceUnregisterResult command

Response Command Name: 'ServiceUnregisterResult'

Description: This is the return command resulting from a ServiceUnregister command. It simply returns the status of the issued ServiceUnregister command.

Arguments:

Arguments only include the **results arguments** detailed in section 2.2.

S-2.4.1.5 ServiceRenewLease command

Command Name: 'ServiceRenewLease'

Description: This is the command used by an ACE service daemon to renew its lease with the ASD. Once this is done, the ASD shall refrain from removing it from it's database for the duration of the lease. As long as the service continues to issue this command it shall not be removed from the ASD.

Arguments:

'name' - The name of the service requesting to renew its lease with the ASD.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'classHierarchy' - The class hierarchy of this service.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING_ARRAY
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'location' - The location name where the service is located.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

'address' - The ACE network address of this service.

Occurrences	: CMDARG_OCCURS_ONCE
Data Type	: CMDVAL_STRING
Allowed Numbers Range	: N/A
Allowed Enum Values	: None
Mutually Dependent	: None

Mutually Exclusive : None

S-2.4.1.6 ServiceRenewLeaseResult command

Response Command Name: 'ServiceRenewLeaseResult'

Description: This is the return command resulting from a ServiceRenewLease command. It simply returns the status of the issued ServiceRenewLease command.

Arguments:

Arguments only include the **results arguments** detailed in section 2.2.

S-2.4.1.7 FlushServices command

Response Command Name: 'FlushServices'

Description: This command is used only by the ASD itself to clear the database of all services. This is used in ASD startup and shutdown.

No Arguments…

S-2.4.1.8 FlushServicesResult command

Response Command Name: 'FlushServicesResult'

Description: This is the return command resulting from a FlushServices command. It simply returns the status of the issued FlushServices command.

Arguments:

Arguments only include the **results arguments** detailed in section 2.2.

S-2.4.1.9 GetServiceLeaseTime command

Response Command Name: 'getServiceLeaseTime'

Description: This command shall inquire with the ASD on how long its service lease time is.

No Arguments…

S-2.4.1.10 GetServiceLeaseTimeResult command

Response Command Name: 'GetServiceLeaseTimeResult'

Description: This is the return command resulting from a GetServiceLeaseTime command. It simply returns the ASD service lease time.

Arguments:

'leaseTime' – The duration of the service lease. That is, for how long the service information shall be kept in the ASD before it is automatically removed by the ASD daemon. This is in milliseconds.

Occurrences	: CMDVAL_OCCURS_ONCE
Data Type	: CMDVAL_INT
Allowed Numbers Range	: { 5000, 3,600,000 }
Allowed Enum Values	: None
Mutually Dependent	: None
Mutually Exclusive	: None

Arguments also include the **results arguments** detailed in section 2.2.

2.4.2 ACE Client to ACE Service Directory Daemon

This section contains the command interface that ACE Service clients use to send commands to the ASD. This section describes the methods and their parameters used to interface with the ASD.

S-2.4.2.1 getServices Method

This method shall query the ASD for services (and their attributes) matching the specific method parameters.

Declaration:

```
public Vector getServices (String name, Vector
                           classHierarchy, String location)
```

Arguments:

`name` – The name of the service we are looking for/querying about.
`classHierarchy` – The vector describing the service class hierarchy that we are looking for. This vector is obtained from the `ACEServiceClassHierarchy` object.
`location` – The name of the location where the service is present.

Return:

A vector of `ACEServiceAttributes` objects for all services found matching the query parameters.

Exception:

`ServiceUnavailableException` – If the ASD is nor running.

S-2.4.2.2 registerService Method

This method shall register a specific service (and its attributes) with the ASD.

Declaration:

```
public void registerService (ACEServiceAttributes
                             servAttribs)
```

Arguments:

`servAttribs` – The `ACEServiceAttributes` of the service we are registering with the ASD.

Return:

None.

Exception:

None.

S-2.4.2.3 unregisterService Method

This method shall remove a specific service (given its attributes) from the ASD.

Declaration:

```
public void unregisterService (ACEServiceAttributes
                               servAttribs)
```

Arguments:

`servAttribs` – The `ACEServiceAttributes` of the service we are unregistering from the ASD.

Return:

None.

Exception:

None.

S-2.4.2.4 renewLease Method

This method shall renew the service lease for a specific service (given its attributes) with the ASD.

Declaration:

```
public void renewLease (ACEServiceAttributes servAttribs)
```

Arguments:

`servAttribs` – The `ACEServiceAttributes` of the service we are renewing the service lease for with the ASD.

Return:

None.

Exception:

None.

S-2.4.2.5 flushDirectory Method

This method shall clear the entire ASD database table.

Declaration:

```
public void flushDirectory ()
```

Arguments:

None.

Return:

None.

Exception:

`ServiceUnavailableException` – If an error occurs sending the command to the ASD.

S-2.4.2.6 getASDLeaseTime Method

This method shall return the lease time allotted for all services that register with the ASD.

Declaration:

```
public long getASDLeaseTime ()
```

Arguments:

None.

Return:

The service lease time given to all services registering with the ASD in milliseconds.

Exception:

None.

S-2.4.2.7 getACEServiceDirectory Method

This static method shall return the command interface to the ASD.

Declaration:

```
public static ACEServiceDirectory getACEServiceDirectory ()
```

Arguments:

None.

Return:

An instance of the command interface to the ASD.

Exception:

`ServiceUnavailableException` – If an error occurs sending the command to the ASD.

3 Design Constraints

<Any known constrains on the design are described here.>

4 Deprecated Specifications

Deprecated Intro

This is a quick and dirty specification for a minimal Ace Service Directory Daemon for use in the KU Engineering Expo 2001. It is to be a rapid prototype that can be thrown away. It only handles a simple 'registration' command and a simple 'list services' command.

5 Glossary

<Refer to a project glossary for terms used throughout the project>

6 Change Log

Version	Date	Changes
0.1	February 20, 2001	Initial Working Version (for EXPO)
0.2	March 13, 2001	Begun expansion towards full-blown ASD.
1.0	May 30, 2001	Initial Released Version

7 Notes

-