

# Language Grid

NICT Language Grid Project

## Language Grid Core Node Specifications

NICT Language Grid Project



# Table of Contents

1. Introduction.....	1
1.1. Provisioning Features of the Language Grid .....	1
2. Web Service Interface.....	2
2.1. Language Service Management API.....	2
2.2. User Management API.....	16
2.3. Access Control Information Management API.....	23
2.4. Access Log Management API.....	28
2.5. Class, Enumerated Type .....	31
3. Access Sites to Language Grid Core Node .....	40
4. Conclusion .....	40
Appendix A.....	41
Translation Service Interface .....	41
Bilingual Dictionary Service Interface .....	42
Parallel Text Search Service Interface .....	43
Morphological Analysis Service Interface.....	44
Paraphrase Service Interface.....	45
Similarity Calculation Service Interface .....	46
Adjacency Pair Search Service Interface .....	47

## 1. Introduction

Language Grid is a multilanguage service base developed by the National Institute of Information and Communications Technology (NICT) that enables users to link language resources on the Internet, such as machine translations or dictionaries, to construct new language services. For example, when a machine translation between two languages is not available, a user can combine several machine translations to create a new desired language pair. A user can also connect a bilingual dictionary with a certain specialty to a machine translation to create a new machine translation for the specified area.

The language grid is composed of two nodes. One, called the Language Grid Service Node, controls language resources and provides an atomic Service, while the other, called the Language Grid Core Node, manages language service registration information, coordinates language services based on workflow and offers composite services. This manual explains the interface for using the Language Grid Core Node.

### 1.1. Provisioning Features of the Language Grid

The Language Grid core node provides the three following services. Two of these, the Language Service Invocation Function and Workflow Execution Function, are invoked by the Core Node when the Language Service is invoked, and only the Information Management Function is invoked externally by a Web service interface.

#### – Information Management Function

Can perform the tasks of information registration, search, editing, and deletion, which are necessary for executing the Language Service

- Language Service information (profile, WSDL for internal invocation, WSDL for external invocation, BPEL)

WSDL for internal invocation is assigned with an atomic service whose access point is on the service node, and WSDL for external invocation is assigned with either an atomic or composite service whose access point is on the core node. Only the WSDL is published for external invocation to enforce access through the core node in service invocation.

- User informaton (organization name, e-mail, home page, address)
- Access limit information (access permission or denial, number of accesses and access limits according to data transfer size)
- Access log information (invocation log, number of accesses/total data transfer size)

#### – Language Service Invocation Function

This function receives a Language Service Invocation request from the user and/or workflow execution function, and invokes the atomic service in the Service Node or the composite service

in the Core Node. Executes the following functions when the Language Service is invoked:

- Access control function  
Controls access to Language Service according to previously registered access control information.
- Logging function
- Service Node Selection function (switches over to Substitute Service Node in the event of impediment)
- Function that transmits copyright information to user (sends copyright information set in Language Resource Profile and license information to user)

– Workflow Execution Function

This function executes BPEL workflow. By extending ActiveBPEL engine, it provides the following functions:

- Use of Language Service Invocation  
In the web service invocation from BPEL workflow, this function uses the Language Service Invocation function to control access according to access control settings registered on the Language Grid.
- Workflow dynamic binding  
This function dynamically binds the atomic service invoked by workflow according to the service ID sent with the service invocation request.

## 2. Web Service Interface

This chapter describes the Web service API for invoking each function in the information management function. For the details of atomic service and composite service execution function API, please see the appendix.

### 2.1. Language Service Management API

#### 2.1.1. Search Services

Assertion	ServiceEntrySearchResult search Services (int startIndex, int maxCount, MatchingCondition[] conditions, Order[] orders, String scope) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UnsupportedMatchingMethodException;	
Argument	startIndex	Start index for search results.
	maxCount	Total number of search results. Maximum is 30.

	conditions	Matching conditions for searching. AND search.
	orders	Sort method of search results
	scope	Range of service for search. (specifies scope value in text)
Return value	Search results for startIndex and maxCount.	
Comment	This method searches the service set in the Language Grid (both external and BPEL services). The range of searches is set by startIndex and maxCount.	
Access rights	This method is available to all users	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedMatchingMethodException	This exception occurs when the available matching method is not specified in the arguments.

### 2.1.2. Search Composite Service

Assertion	CompositeServiceEntrySearchResult searchCompositeServices (int startIndex, int maxCount, MatchingCondition[] conditions, Order[] orders, String scope) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UnsupportedMatchingMethodException;	
Argument	startIndex	Start index of search results.
	maxCount	Total number of search results. Maximum is 30

	conditions	Matching condition for searching. AND search.
	orders	Sort method of search results
	scope	Range of service for search. (specifies scope value in text)
Return value	Search results for startIndex and maxCount.	
Comment	Searches services registered on the Language Grid (BPEL Service only). The results include atomic service information used by that service. The range of searches is set by startIndex and maxCount.	
Access rights	This method is available to all users	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocation exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedMatchingMethodException	This exception occurs when the available matching method is not specified in the arguments.

### 2.1.3. Add Services

Assertion	void addService (String serviceId, ServiceProfile profile, ServiceInstance instance, Attribute[] attributes)  throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceAlreadyExistsException, ServiceConfigurationException, UnknownException;	
Argument	serviceId	Service ID
	profile	Information about service profile
	instance	Service instance
	attributes	Attributes of services

Return value	None	
Comment	<p>This method registers a language service in the Language Grid. In registering, a provider can choose a service ID freely, but if the service ID has already been in use, this method returns ServiceAlreadyExistsException.</p> <p>This method enables a provider to register the following two types of services.</p> <ul style="list-style-type: none"> <li>● External Services</li> </ul> <p>These services are deployed on Language Grid Service Node and on servers outside the Language Grid. Users can specify the jar compress file, which consists of WSDL for the service (file name is “ServiceID.wsdl”) as an instance (“instance.instance”).</p> <ul style="list-style-type: none"> <li>● BPEL Composite Services</li> </ul> <p>This is a complex service based on BPEL. Users can specify the jar compress file, which consists of BPEL file (“serviceID.bpel”) and its WSDL files (“arbitrary name.wsdl”) as an instance (instance.instance).</p>	
Access rights	This method is available to all users other than anonymous users.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceAlreadyExistsException	This exception occurs when the same service ID is set.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails by unknown reasons.

#### 2.1.4. Delete Service

Assertion	void deleteService (String serviceId) throws           AccessLimitExceededException,           InvalidParameterException, NoAccessPermissionException,                           ServiceConfigurationException, ServiceNotFoundException, ServiceNotInactiveException, UnknownException;	
Argument	serviceId	Service ID
Return value	None	
Comment	This method deletes services that are already registered in the Language Grid.	
Access rights	This method is available to users who have registered the specified services in the	

	Language Grid	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.5. Acquire Service Profile

Assertion	ServiceProfile getServiceProfile (String serviceId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
Return value	Service profile information	
Comment	This method obtains service profiles that have been registered in the Language Grid.	
Access rights	This method is available to all users	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception is occurred when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.6. Set Service Profile

Assertion	void setServiceProfile(String serviceId, ServiceProfile profile) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	profile	Information about service profile.
Return value	None	
Comment	This method sets the service profile that has been registered in the Language Grid.	
Access rights	This method is available to users who have had registered the specified services in the Language Grid.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.7. Acquire Service Attribute

Assertion	Attribute[] getServiceAttributes (String serviceId, String[] attributeNames) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	attributeNames	Array of attribute names.
Return value	Array of attributes	
Comment	This method returns information about service attributes registered in the Language Grid. Users can specify the array of attribute names at “attributeName”. When users specify a non-existing attribute name, the system ignores it. When users specify an empty array, they can get all attributes.	

Access rights	This method is available to users who have registered the specified services in the Language Grid.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.8. Set Service Attribute Information

Assertion	void setServiceAttributes (String serviceId, Attribute[] attributes) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	attributes	Array of attributes
Return value	None	
Comment	This method sets the information about service attributes registered in the Language Grid. A user can specify the array of attributes in “attributes”. When they specify the entire value in the attribute, the system deletes the attribute.	
Access rights	This method is available to users who have registered the specified services in the Language Grid.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.

	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.9. Acquire Service Instance

Assertion	ServiceInstance getServiceInstance (String serviceId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
Return value	Information about service instances.	
Comment	This method obtains the service instances registered in the Language Grid.	
Access rights	This method is available to users who have registered the specified services in the Language Grid.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.10. Set Service Instance

Assertion	void setServiceInstance (String serviceId, ServiceInstance instance) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID

	instance	Information about a service instance
Return value	None	
Comment	This method sets information about a service instance registered in the Language Grid. If the system invokes this API when the service is active, and then “ServiceNotInactiveException” occurs, users should deactivate the service by using “deactivateService API” before invoking this API.	
Access rights	This method is available to users who have registered the specified services in the Language Grid.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	ServiceNotInactiveException	This exception occurs when the service is not active.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.11.Acquire WSDL

Assertion	byte[] getServiceWsd (String serviceId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
Return value	WSDL of the service.	
Comment	Acquires WSDL to invoke services registered on Language Grid	
Access rights	This method is available to all users.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.

	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.12. Activate Services

Assertion	void activateService(String serviceId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotActivatableException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
Return value	None	
Comment	This method activates services registered in the Language Grid. When a service is already active, the method does nothing. When the method fails, the “ServiceNotActivatableException” occurs.	
Access rights	This method is available to users who have registered the specified services in the Language Grid	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotActivatableException	This exception occurs when the service is not active.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.

	UnknownException	This exception occurs when the service fails for unknown reasons.
--	------------------	---

### 2.1.13.Deactivate Services

Assertion	void deactivateService (String serviceId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotActivatableException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
Return value	None	
Comment	This method inactivates a service registered in the Language Grid inactive. When the service is inactive, the system does nothing. When the system fails, the “ServiceNotDeactivatableException” occurs.	
Access rights	This method is available to users who have the specified services added to their accounts.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocation exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotDeactivatableException	This exception occurs when the service is active.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.1.14.List Service Endpoints

Assertion	Endpoint[] listActualEndpoints (String serviceId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID

Return value	An array of service endpoints	
Comment	This method lists endpoints of a specified service ID.	
Access rights	This method is available to users who register a service ID.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedMatchingMethodException	This exception occurs when the available matching method is not specified in the arguments.

### 2.1.15.Add Service Endpoints

Assertion	void addActualEndpoint (String serviceId, Endpoint endpoint) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	endpoint	Endpoint associated with the service ID
Return value	None	
Comment	This method associates an endpoint with a specified service ID. The endpoint consists of a url and a userName. When the specified endpoint has already been associated with the service ID, InvalidParameterException occurs.	
Access rights	This method is available to users who register a service ID	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.

	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedMatchingMethodException	This exception occurs when the available matching method is not specified in the arguments.

### 2.1.16.Delete Service Endpoint

Assertion	void deleteActualEndpoint (String serviceId, String url, String userName) throws AccessLimitExceededException, EndpointNotFoundException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	url	URL of endpoint
	userName	User name to be authenticated
Return value	None	
Comment	This method eliminates an association between the specified endpoint and the specified service ID.	
Access rights	This method is available to users who register a service ID	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	EndpointNotFoundException	This exception occurs when the method cannot find the specified endpoint.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.

	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedMatchingMethodException	This exception occurs when the available matching method is not specified in the arguments.

### 2.1.17.Enable Service Endpoint

Assertion	void enableEndpoint (String serviceId, String url, String userName) throws AccessLimitExceededException, EndpointNotFoundException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	url	URL of an endpoint
	userName	User name to be authenticated
Return value	None	
Comment	This method enables a specified endpoint.	
Access rights	This method is available to users who register a service ID	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	EndpointNotFoundException	This exception occurs when the method cannot find the specified endpoint.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.

	UnknownException	This exception occurs when the service fails for unknown reasons.
--	------------------	---

### 2.1.18. Disable Service Endpoint

Assertion	void disableEndpoint (String serviceId, String url, String userName) throws AccessLimitExceededException, EndpointNotFoundException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, ServiceNotFoundException, UnknownException;	
Argument	serviceId	Service ID
	url	URL of an endpoint
	userName	User name to be authenticated
Return value	None	
Comment	This method disables the specified endpoint. Access to the disabled endpoint is not permitted.	
Access rights	This method is available to users who register a service ID	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	EndpointNotFoundException	This exception occurs when the method cannot find the specified endpoint.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotFoundException	This exception occurs when the method cannot find the specified service ID.
	UnknownException	This exception occurs when the service fails for unknown reasons.

## 2.2. User Management API

### 2.2.1. Search User

Assertion	UserEntrySearchResult searchUsers (int startIndex, int maxCount, MatchingCondition[] conditions, Order[] orders) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException,	
-----------	---	--

	UnknownException;	
Argument	startIndex	Start index of search results.
	maxCount	Total number of search results. Maximum is 30
	conditions	Matching condition for searching. AND search.
	orders	Search information sort method
Return value	Search results for startIndex and maxCount.	
Comment	This method searches user data registered in the Language Grid.	
Access rights	This method is available only to an administrator user.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedMatchingMethodException	This exception occurs when the available matching method is not specified in the arguments.

### 2.2.2. Add User

Assertion	void addUser (String userId, String password, UserProfile profile, Attribute[] attributes) throws AccessLimitExceededException, InvalidParameterException, InvalidUserIdException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserAlreadyExistsException;	
Argument	userId	User ID
	profile	User profile to be registered in the language Grid

	attributes	Extended attributes of user information
Return value	None.	
Comment	This method is available only to an administrator user.	
Access rights	This method adds a user to the Language Grid.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	InvalidUserIdException	This exception occurs when the service adds an invalid user ID.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserAlreadyExistsException	This exception occurs when the specified user ID is in use.

### 2.2.3. Delete User

Assertion	void deleteUser (String userId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserNotFoundException;	
Argument	userId	User ID
Return value	None.	
Comment	This method deletes a user from a list of users registered in the Language Grid	
Access rights	This method is available only to an administrator user.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.

	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.

#### 2.2.4. Acquire User Profile Information

Assertion	UserProfile getUserProfile (String userId) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserNotFoundException;	
Argument	userId	User ID
Return value	User profile information	
Comment	This method acquires the profile information of the specified user.	
Access rights	This method is available only to an administrator or user himself.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.

#### 2.2.5. Set User Profile Information

Assertion	void setUserProfile (String userId, UserProfile profile) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException,	
-----------	--	--

	UnknownException, UserNotFoundException;	
Argument	userId	User ID
	profile	Profile information to be set
Return value	None	
Comment	This method sets the user profile information of a specified user.	
Access rights	This method is available only to an administrator or user himself	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.

### 2.2.6. Acquire Extended User Attributes

Assertion	Attribute[] getUserAttributes (String userId, String[] attributeNames) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserNotFoundException;	
Argument	serviceId	Service ID
	attributeNames	Array of attribute names.
Return value	Array of attributes	
Comment	<p>This method acquires the attribute information of a specified user and returns the values of the attributes specified by the attributeNames argument. Users can specify the array of attribute names at “attributeName”.</p> <p>When users specify a non-existing attribute name, the system ignores it. When users specify an empty array, they can get all attributes.</p>	
Access rights	This method is available only to an administrator or user himself.	

Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.

### 2.2.7. Set Extended User Attributes

Assertion	void setUserAttributes (String userId, Attribute[] attributes) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserNotFoundException;	
Argument	userId	User ID
	attributes	Array of attributes
Return value	None	
Comment	This method sets the extended user attributes of the specified user. A user can specify the array of attributes in “attributes”. When they specify the entire value in the attribute, the system deletes the attribute.	
Access rights	This method is available only to an administrator or user himself.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocation exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.
--	-----------------------	---

### 2.2.8. Acquire Last Update of User Password

Assertion	Calendar getPasswordChangedDate (String userId) throws           AccessLimitExceededException,           InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserNotFoundException;	
Argument	userId	User ID
Return value	Password change date	
Comment	Acquires user password change date	
Access rights	This method is available only to an administrator or user himself.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.

### 2.2.9. Change User Password

Assertion	void setUserPassword (String userId, String password) throws           AccessLimitExceededException,           InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UserNotFoundException;	
Argument	userId	User ID
	password	Password
Return value	None	
Comment	This method changes the password of a specified user.	
Access rights	This method is available only to an administrator or user himself.	

Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UserNotFoundException	This exception occurs when the service cannot find the specified user ID.

## 2.3. Access Control Information Management API

### 2.3.1. Search Access Permission

Assertion	<code>AccessRightSearchResult searchAccessRights (int startIndex, int maxCount, String userId, String serviceId, Order[] orders)</code> throws <code>AccessLimitExceededException</code> , <code>InvalidParameterException</code> , <code>NoAccessPermissionException</code> , <code>ServiceConfigurationException</code> , <code>UnknownException</code> ;	
Argument	startIndex	Start index of search results.
	maxCount	Total number of search results. Maximum is 30.
	userId	User ID. "" when omitted.
	serviceId	Service ID. "" when omitted.
	orders	Search information sort method
Return value	Search results for startIndex and maxCount.	
Comment	This method searches access permission settings of a specified service and a specified user. It returns an array of results whose number is specified by maxCount and whose first element has an index number specified by startIndex or following startIndex.	
Access rights	This method is available to all users other than anonymous users. Users can invoke this method for the services for which they have registered. When Service ID is abbreviated, search is performed from accessible services.	

Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.3.2. Set Access Permission

Assertion	void setAccessRight (String userId, String serviceId, boolean permitted, Calendar startDateTime, Calendar endDateTime) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException;	
Argument	userId	User ID
	serviceId	Service ID
	permitted	true: Permitted, false: Denied
	startDateTime	Date and time this setting started
	endDateTime	Date and time of this setting ended
Return value	None	
Comment	Sets access rights. Access rights are determined according to User ID and Service ID. If the same setting already exists, the setting is overwritten. The service regulation value can be set by specifying "*" in the User ID.	
Access rights	This method is available to all users other than anonymous users. Users can invoke this method for the services for which they have registered.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.

	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.3.3. Delete Access Permission

Assertion	void deleteAccessRight (String userId, String serviceId) throws AccessLimitExceededException, AccessRightNotFoundException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException;	
Argument	userId	User ID
	serviceId	Service ID
Return value	None	
Comment	This method deletes the access permission setting of the specified service.	
Access rights	This method is available to all users other than anonymous users. Users can invoke this method for the services for which they have registered.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	AccessRightNotFoundException	This exception occurs when the method cannot find the access permission setting of the specified service.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.3.4. Access Limit Information Search

Assertion	AccessLimitSearchResult searchAccessLimit (int startIndex, int maxCount, String
-----------	---

	userId, String serviceId, Order[] orders) throws          AccessLimitExceededException,          InvalidParameterException, NoAccessPermissionException,          ServiceConfigurationException, UnknownException;	
Argument	startIndex	Start index of search results.
	maxCount	Total number of search results. Maximum is 30.
	userId	User ID. "" when omitted.
	serviceId	Service ID. "" when omitted.
	orders	Search information sort method
Return value	Among the search results, results from range specified by startIndex and maxCount	
Comment	Searches access limit information for specified user and service.	
Access rights	Accessible to all but anonymous users. When service ID is abbreviated, searches accessible services. Services registered by the users invoking this API are targeted for search.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.3.5. Access Limit Information Setting

Assertion	void setAccessLimit (String userId, String serviceId, Period period, LimitType limitType, int limitCount, Calendar startDateTime, Calendar endDateTime) throws          AccessLimitExceededException,          InvalidParameterException, NoAccessPermissionException,          ServiceConfigurationException, UnknownException;	
Argument	userId	User ID
	serviceId	Service ID

	period	Period of time
	limitType	Ttype of limit
	limitCount	Limit value
	startDateTime	Date and time this setting starting
	endDateTime	Date and time this setting ended
Return value	None	
Comment	Sets access control information. Access control is determined by user and service ID, control type and period. If the same setting already exists, the setting is overwritten. The service regulation value can be set by specifying "*" in the User ID.	
Access rights	Accessible to all but anonymous users. Services registered by the users invoking this API are targeted for search.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.3.6. Access Control Information Deletion

Assertion	void deleteAccessLimit (int accessLimitId) throws AccessLimitExceededException, AccessRightNotFoundExcep- tion, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException;	
Argument	accessLimitId	Access control information ID
Return value	None	
Comment	Deletes access permission information.	
Access rights	Accessible to all but anonymous users. Services registered by the users invoking this API are targeted for search.	

Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	AccessRightNotFoundException	This exception occurs when the method cannot find the access permission setting of the specified service.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

## 2.4. Access Log Management API

### 2.4.1. Search Access Log

Assertion	<p>AccessLogSearchResult searchAccessLog (int startIndex, int maxCount, String userId, String serviceId, Calendar startDateTime, Calendar endDateTime, Order[] orders)</p> <p>throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException;</p>	
Argument	startIndex	Start index of search results
	maxCount	Total number of search results. Maximum is 30.
	userId	User ID. "" when omitted.
	serviceId	Service ID. "" when omitted.
	startDateTime	Start date and time of access logs. Access logs include the start date and time.
	endDateTime	End date and time of access logs. Access logs do not include the end date and time.
	orders	Search information sort method
Return	The range of searches is set by startIndex and maxCount.	

value		
Comment	This method searches access logs of a specified service invoked by a specified user. It returns an array of results whose number is specified by maxCount and whose first element has an index number specified by startIndex or following startIndex. When serviceId is set to "", this method searches the access logs of all the services for that the invoker registered.	
Access rights	This method is available to all users other than anonymous users. Users can invoke this method for the services for which they have registered.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

#### 2.4.2. Acquisition of Number of Accesses by Period

Assertion	<code>int[] getAccessCounts (String userId, String serviceId, Calendar baseDateTime, String period)</code> throws <code>AccessLimitExceededException</code> , <code>InvalidParameterException</code> , <code>NoAccessPermissionException</code> , <code>ServiceConfigurationException</code> , <code>UnknownException</code> ;	
Argument	userId	User ID. "" when omitted.
	serviceId	Service ID. "" when omitted.
	baseDateTime	Standard date and time
	period	Period (supports only DAY, MONTH, YEAR)
Return value	Number of accesses. When period is DAY, int[24], when MONTH, int[31], and when YEAR, int[12]	
Comment	Totals monthly accesses for specified year.	
Access rights	Accessible to all but anonymous users. Services registered by users invoking this API are targeted for search.	

Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

### 2.4.3. Access Totalling

Assertion	<p>UserAccessEntrySearchResult sumUpUserAccess (int startIndex, int maxCount, String serviceId, Calendar startDateTime, Calendar endDateTime, String period, Order[] orders)</p> <p>throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, ServiceConfigurationException, UnknownException;</p>	
Argument	startIndex	Start index of search results
	maxCount	Total number of search results. Maximum is 30.
	serviceId	Service ID
	startDateTime	Search start date and time. This date and time is included in search.
	endDateTime	Search end date and time. This date and time is included in search.
	period	Period (supports only DAY, MONTH, and YEAR)
	orders	Search information sort method
Return value	Total accesses for a specified period.	
Comment	Totals the accesses for a specified period.	
Access rights	Accessible to all but anonymous users. Services registered by users invoking this API are targeted for search.	

Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	UnknownException	This exception occurs when the service fails for unknown reasons.

## 2.5. Class, Enumerated Type

The classes and the enumerated type of APIs mentioned above are as follows. In the enumerated type, the defined identifiers are saved as text. Therefore, they are used as a String-type in the web service interface.

### 2.5.1. ServiceEntrySearchResult

**Field:**

```
ServiceEntry[] elements; //sequence of entry
int totalCount; //Number of entries hit by a search operation
boolean totalCountFixed; //Boolean of the fix of the number of hit entries
```

**Comment:**

Information about ServiceEntrySearchResult is obtained when the system invokes searchServices. When the number of hit entries is not fixed in one invocation (ex. Complex search), CountFixed is false. In this case, when a user wants it to be true, the system should continue searching using the same conditions.

### 2.5.2. ServiceEntry

**Field:**

```
String serviceId; //Service ID.
String displayName; //Service display name. (about 50 characters per line)
ServiceType serviceType; //Service type
LanguagePath[] supportedLanguages; //Language
String endpointUrl; // Endpoint URL
String ownerUserId; //User ID of registered member
```

Calendar registeredDate; //Date of registration

Calendar updatedDate; //Date of update

boolean active; //Boolean of active Service

**Comment:**

Information about ServiceEntrySearchResult is obtained when the system invokes searchServices. It consists of part of the information about ServiceProfile and ServiceInstance and the information about the system that is saved automatically.

### 2.5.3. CompositeServiceEntrySearchResult

**Field:**

CompositeServiceEntry[] elements; //sequence of entry

int totalCount; //Number of entries retrieved by a search operation

boolean totalCountFixed; //Boolean of the fix of the number of hit entries

**Comment:**

This is information acquirable at time of composite service search (searchCompositeServices). When the number of hit entries is not fixed in one call (ex. Complex search), CountFixed is false. In this case, when a user wants it to be true, the system should continue searching using the same conditions.

### 2.5.4. CompositeServiceEntry

**Field:**

String serviceId; //Service ID

String displayName; //Service display name. (about 50 characters per line)

ServiceType serviceType; //Service type

LanguagePath[] supportedLanguages; //Language

String endpointUrl; //Endpoint URL

String ownerUserId; //User ID of registered member

Calendar registeredDate; //Date of registration

Calendar updatedDate; //Date of update

boolean active; //Boolean of active Service

ServiceEntry[] servicesInUse; // atomic service that uses this composite service

**Comment:**

This is information acquirable at time of composite service search (searchCompositeServices). Added to part of the information stored in ServiceProfile and ServiceInstance, it stores information automatically set by system and atomic service information used by composite service.

### 2.5.5. ServiceProfile

**Field:**

String displayName; //Service display name. (about 50 characters per line)  
String serviceProviderName; //Name of service provider  
String serviceProviderUri; //URL of service provider  
String registerName; //Name of registration member  
String registerUri; //URL of registration member  
String copyrightInfo; //Copyright information  
String contactName; //contact name  
String contactUri; //contact URI  
String contactEmail; //contact email address

**Comment:**

This is information about the service profile. It consists of the information a user can modify in the Language Grid service information.

### 2.5.6. ServiceInstance

**Field:**

ServiceType serviceType; // Service type of the service instance  
LanguagePath[] supportedLanguages; // Language path supported by the service instance  
InstanceType instanceType; // Type of service instance  
String instanceSize; // Binary data size of the service instance  
byte[] instance; // Binary data of the service instance

**Comment:**

This is information about the service instance, which is defined as data necessary to run the language service and is managed by using the Language Grid. The binary data of the service instance varies depending on the type of service instance. Please refer to section 5.1.2 Add Services for more details.

### 2.5.7. ServiceType (Enumerated Type)

**Value of enumeration:**

ADJACENCYPAIR //adjacency pair  
BACKTRANSLATION //back translation  
BILINGUALDICTIONARY //bilingual dictionary  
DICTIONARY //dictionary

MORPHOLOGICALANALYSYS //morphological analysis  
MULTIHOPTRANSLATION //multi hop translation  
PARALLELTEXT //parallel text  
PARAPHRASE //paraphrase  
SIMILARITYCALCULATION //similarity calculation  
TRANSLATION //translation  
OTHER //others

**Comment:**

This is the enumerated type that specifies service type.

### 2.5.8. LanguagePath

**Field:**

String[] languages; //sequence of language expressions

**Comment:**

This saves the language path. Each language expression obeys RFC 3066.

### 2.5.9. InstanceType (Enumerated Type)

**Value of enumeration:**

EXTERNAL //External service (atomic service)  
BPEL //BPEL service (complex service)

**Comment:**

This is an enumerated type that defines the type of service.

### 2.5.10. MatchingCondition

**Field:**

String fieldName; //Field name  
String matchingValue; //Matching value with field  
String matchingMethod; //Matching method (MatchingMethod value shown as text)

**Comment:**

This consists of service and user's search criteria. MatchingField specifies the target data field (ServiceEntry, UserEntry) and target attribute (getServiceAttributes/getUserAttributes).

### 2.5.11. MatchingMethod (Enumerated Type)

**Value of enumeration:**

COMPLETE //Complete matching  
PARTIAL //Partial matching

PREFIX //Prefix matching  
SUFFIX //Suffix matching  
REGEX //Regular expression matching

**Comment:**

This defines the matching method between query and the data during the search. Which values are supported depends on the invoked services.

## 2.5.12.Order

**Field:**

String fieldName; //Field name  
String direction; // Resort direction (OrderDirection value shown as text)

**Comment:**

Specifies resort method for searches returned by API. fieldName is specifiable for each search subject data field (ServiceEntry, UserEntry).

## 2.5.13.OrderDirection (Enumerated Type)

**Value of enumeration:**

ASCENDANT // Ascendant  
DESCENDANT // Descendant

**Comment:**

Enumeration type that defines resort method for returned search results at time of search.

## 2.5.14.Scope (Enumerated Type)

**Value of enumeration:**

ALL // All services  
MINE // User registration service executing search  
ACCESSIBLE // Service accessible to user performing search

**Comment:**

Enumeration type that defines range of services subject to search at time of search. When specified as ACCESSIBLE, results are returned according to accessibility (AccessRight). Whether access limit (AccessLimit) is exceeded is not considered.

## 2.5.15.Attribute

**Field:**

String name; //Name of attribute  
String value; //Value of attribute

**Comment:**

Saves the service and user's information.

## 2.5.16.Endpoint

**Field:**

String url; //URL  
String userName; //Certification user name  
String password; //Certification password  
boolean enabled; //Boolean of the enabled endpoint

**Comment:**

This method saves the endpoint information.

## 2.5.17.UserEntrySearchResult

**Field:**

UserEntry[] elements; //Array of user entry  
int totalCount; //Total number of search results  
boolean totalCountFixed; //Boolean of the fix of the number of hit entries

**Comment:**

This method saves the user's search results.

## 2.5.18.UserEntry

**Field:**

String userId; //User ID  
String organization; //Organization  
String representative; // Responsible Person  
String emailAddress; // E-mail Address  
String homepageUrl; // Homepage URL  
String address; // Address  
Calendar registeredDate; //Registered Date  
Calendar updatedDate; //Update Date

**Comment:**

This method saves the user's entry information.

## 2.5.19.UserProfile

**Field:**

String organization; //Organization

```
String representative; // Responsible Person
String emailAddress; // E-mail Address
String homepageUrl; // Homepage URL
String address; // Address
```

**Comment:**

This method saves the user's profile.

## 2.5.20.AccessRightSearchResult

**Field:**

```
AccessRight[] elements; //Array of permission
int totalCount; //Number of search results
boolean totalCountFixed; //Boolean of the fix of the number of hit entries
```

**Comment:**

This method saves the search results for the user's permissions.

## 2.5.21.AccessRight

**Field:**

```
String userId; //User ID
String serviceId; //Service ID
boolean permitted; //Boolean of user's permission
Calendar startDateTime; // Start date and time
Calendar endDateTime; // end date and time
```

**Comment:**

This method saves the information about the user's permission

## 2.5.22.Period (Enumerated Type)

**Value of enumeration:**

```
EACHTIME //One access
DAY // one day
MONTH // one day
YEAR // one year
```

**Comment:**

Enumeration type that defines access total period when setting access limit.

## 2.5.23.LimitType (Enumerated Type)

**Value of enumeration:**

FREQUENCY // frequency

CAPACITY // capacity

**Comment:**

Enumeration type that defines access limit type when setting access limit.

## 2.5.24. AccessLimitSearchResult

**Field:**

AccessLimit[] elements; //Array of permission

int totalCount; //Number of search results

boolean totalCountFixed; //Boolean of the fix of the number of hit entries

**Comment:**

This method saves the search results for the user's permissions.

## 2.5.25. AccessLimit

**Field:**

int accessLimitId; // limit ID

String userId; //User ID

String serviceId; //Service ID

String period; // Period

String limitType; // limit type (LimitType)

int limitCount; // limit value

Calendar startDateTime; // start date and time

Calendar endDateTime; // end date and time

**Comment:**

Stores access limit information

## 2.5.26. AccessLogSearchResult

**Field:**

AccessLog[] elements; //array of access log

int totalCount; //Number of search results

boolean totalCountFixed; // whether or not actually hit entries are established

**Comment:**

This method saves the access logs search results.

## 2.5.27. AccessLog

**Field:**

```
String address; //Original access address
String host; //Original access host machine name
Calendar dateTime; //Access date
String requestUri; //Target access URI
int responseCode; //Response code
int responseBytes; //Response size
String referer; //HTTP referer
String agent; // HTTPagent
String faultCode; //fault SOAP code
String faultString; //fault SOAP strings
String userId; //User ID
String serviceId; //Service ID
String nodeId; //Node ID (out of service)
```

**Comment:**

This method saves access log information.

## 2.5.28. UserAccessEntrySearchResult

**Field:**

```
UserAccessEntry[] elements; //Array of user entry
int totalCount; //Total number of search results
boolean totalCountFixed; //Boolean of the fix of the number of hit entries
```

**Comment:**

Stores access total results.

## 2.5.29. UserAccessEntry

**Field:**

```
String userId; //User ID
String userOrganization; // user organization
String serviceId; //Service ID
String serviceName; // Service name
int accessCount; // Number of accesses
int transferredSize; // Forwarded data amount
```

**Explanation:**

Stores access total information

### 3. Access Sites to Language Grid Core Node

The Language Grid core node is currently being operated by NICT. When sending Language Grid User ID, we also send the URL of the Language Grid Core Node. When accessing services, please get WSDL files for each interface from the URL and produce a stub from the WSDL.

In the future, the Language Grid will be distributed using the P2P Grid. When being distributed, interfaces such as the computation resource management interface will be added.

### 4. Conclusion

Please contact the following address if you have any questions about this manual.

Contact: [langrid@khn.nict.go.jp](mailto:langrid@khn.nict.go.jp)

## Appendix A

### Translation Service Interface

Assertion	String translate (String sourceLang, String targetLang, String source) throws AccessLimitExceededException, InvalidParameterException, LanguagePairNotUniquelyDecidedException, NoAccessPermissionException, ServiceConfigurationException, TranslationFailedException, UnknownException, UnsupportedLanguagePairException;	
Argument	sourceLang	Source language
	targetLang	Target language
	source	Sentence to be translated.
Return value	Translation result	
Comment	This translates a sentence in a source language into a target language. The source and target languages are specified in the arguments, sourceLang and targetLang, respectively. If the language pair is not uniquely decided, this throws LanguagePairNotUniquelyDecidedException. To avoid this exception, it is necessary to specify a more concrete language code for the arguments. (e.g. (“jp”, “zh”) -> (“jp”, “zh-CN”))	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	LanguagePairNotUniquelyDecidedException	This exception occurs when the language pair is not uniquely decided.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotActiveException	This exception occurs when the invoked service is not active.
	TranslationFailedException	This exception occurs when the invoked service fails to translate.

	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguagePairException	This exception occurs when the available language pair is not specified in the arguments.

## Bilingual Dictionary Service Interface

Assertion	Translation[] search (String headLang, String targetLang, String headWord, String matchingMethod) throws AccessLimitExceededException, InvalidParameterException, LanguagePairNotUniquelyDecidedException, NoAccessPermissionException, SearchFailedException, ServiceConfigurationException, UnknownException, UnsupportedLanguagePairException;	
Argument	headLang	Language of head word
	targetLang	Language of translation
	headWord	Head word to be looked up in a bilingual dictionary
	matchingMethod	Matching method (Value of MatchingMethod)
Return value	Search results	
Comment	This method looks up a word in a bilingual dictionary by language and head word. In searching the bilingual dictionary, it uses the matching method specified by the matchingMethod argument and returns the translation of the head word.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	LanguagePairNotUniquelyDecidedException	This exception occurs when the language pair is not decided uniquely.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	SearchFailedException	This exception occurs when the service fails to look up the word in the bilingual dictionary.

	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotActiveException	This exception occurs when the invoked service is not active.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguagePairException	This exception occurs when the available language pair is not specified in the arguments.

## Parallel Text Search Service Interface

Assertion	ParallelText[] search (String sourceLang, String targetLang, String text, String matchingMethod) throws AccessLimitExceededException, InvalidParameterException, LanguagePairNotUniquelyDecidedException, NoAccessPermissionException, SearchFailedException, ServiceConfigurationException, UnknownException, UnsupportedLanguagePairException;	
Argument	sourceLang	Language of an input sentence
	targetLang	Language of translation
	source	Input sentence
	matchingMethod	Matching method (Value of MatchingMethod)
Return value	Search results	
Comment	This method searches translation of an input text. For the search, this function uses the matching method specified in the matchingMethod argument and returns a translation corresponding to the input sentence.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when this service is invoked with an invalid argument.
	LanguagePairNotUniquelyDecidedException	This exception occurs when the language pair is not uniquely decided.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.

	SearchFailedException	This exception occurs when the service fails to look up the input sentence in the parallel texts.
	ServiceConfigurationException	This exception occurs when this service is not configured correctly.
	ServiceNotActiveException	This exception occurs when the invoked service is not active.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguagePairException	This exception occurs when the available language pair is not specified in the arguments.

## Morphological Analysis Service Interface

Assertion	Morpheme[] analyze (String language, String text) throws AccessLimitExceededException, AnalysisFailedException, InvalidParameterException, LanguageNotUniquelyDecidedException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UnsupportedLanguageException;	
Argument	language	Language of a sentence to be analyzed
	text	Sentence to be analyzed
Return value	Analysis result	
Comment	This method executes morphological analysis.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	AnalysisFailedException	This exception occurs when the service fails to analyze an input sentence.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	LanguageNotUniquelyDecidedException	This exception occurs when the language is not uniquely decided.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.

	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotActiveException	This exception occurs when the invoked service is not active.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguageException	This exception occurs when the available language is not specified in the arguments.

### Paraphrase Service Interface

Assertion	String[] paraphrase (String language, String text) throws AccessLimitExceededException, InvalidParameterException, LanguageNotUniquelyDecidedException, NoAccessPermissionException, ParaphraseFailedException, ServiceConfigurationException, UnknownException, UnsupportedLanguageException;	
Argument	language	Language of a sentence to be paraphrased
	text	Sentence to be paraphrased
Return value	Paraphrase result	
Comment	This method paraphrases an input text.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	LanguageNotUniquelyDecidedException	This exception occurs when the language is not uniquely decided.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	ParaphraseFailedException	This exception occurs when the service fails to paraphrase the input sentence.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.
	ServiceNotActiveException	This exception occurs when the invoked service is not active.

	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguageException	This exception occurs when the available language is not specified in the arguments.

## Similarity Calculation Service Interface

Assertion	double calculate (String language, String text1, String text2) throws AccessLimitExceededException, CalculationFailedException, InvalidParameterException, LanguageNotUniquelyDecidedException, NoAccessPermissionException, ServiceConfigurationException, UnknownException, UnsupportedLanguageException;	
Argument	language	Language of sentences between which similarity is calculated
	text1	Sentence whose similarity to text2 is calculated
	text2	Sentence whose similarity to text1 is calculated
Return value	Similarity. Value between 0 and 1 (1 means the languages are the same.)	
Comment	This method calculates the similarity between text1 and text2 and returns the similarity value.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	LanguageNotUniquelyDecidedException	This exception occurs when the language is not uniquely decided.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	CalculationFailedException	This exception occurs when the service fails to calculate the similarity between input sentences.
	ServiceConfigurationException	This exception occurs when the service is not configured correctly.

	ServiceNotActiveException	This exception occurs when the invoked service is not active.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguageException	This exception occurs when the available language is not specified in the arguments.

## Adjacency Pair Search Service Interface

Assertion	AdjacencyPair[] search (String category, String language, String firstTurn, String matchingMethod) throws AccessLimitExceededException, InvalidParameterException, NoAccessPermissionException, LanguageNotUniquelyDecidedException, SearchFailedException, ServiceNotActiveException, ServiceConfigurationException, UnknownException, UnsupportedLanguageException;	
Argument	category	Dialog category
	language	Language in a dialogue
	firstTurn	Request in a dialogue
	matchingMethod	Matching method (Value of matchingMethod)
Return value	Search results	
Comment	This method searches responses to the input request and returns a list of responses.	
Exception	AccessLimitExceededException	This exception occurs when the number of service invocations exceeds an access limit.
	InvalidParameterException	This exception occurs when the service is invoked with an invalid argument.
	LanguageNotUniquelyDecidedException	This exception occurs when the language is not uniquely decided.
	NoAccessPermissionException	This exception occurs when the service is invoked without access permission.
	SearchFailedException	This exception occurs when the service fails to search the input request in a set of adjacency pairs.
	ServiceNotActiveException	This exception occurs when the service is not configured correctly.

	ServiceConfigurationException	This exception occurs when the invoked service is not active.
	UnknownException	This exception occurs when the service fails for unknown reasons.
	UnsupportedLanguageException	This exception occurs when the available language is not specified in the arguments.