Phase 2 - Identity Services
Marketing Requirements Document

Version: 1.0

NOTE: This document is published for historical interest. Market Requirements Documents were originally not designed for publication outside of the Liberty Alliance membership, and so may not be readily understandable without prior knowledge of the technical terms used by the groups that wrote them. This MRD was developed by the Business Marketing Expert Group (BMEG) and passed on to the Technology Expert Group (TEG) as guiding input to the technical specification development process. As with any development process, changes were made to the list of requirements that the specifications were developed to meet during the course of their development. These changes were agreed to by both BMEG and TEG but were not folded back into this MRD, so there are some discrepancies between the contents of this document and the related technical specifications. Changes to the requirements were made for a number of reasons, both technical and market-related.

Notice:

This document has been prepared by Sponsors of the Liberty Alliance. Permission is hereby granted to use the document solely for the purpose of implementing the Specification. No rights are granted to prepare derivative works of this Specification. Entities seeking permission to reproduce portions of this document for other uses must contact the Liberty Alliance to determine whether an appropriate license for such use is available.

Implementation of certain elements of this document may require licenses under third party intellectual property rights, including without limitation, patent rights. The Sponsors of and any other contributors to the Specification are not, and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights. **This Specification is provided "AS IS", and no participant in the Liberty Alliance makes any warranty of any kind, express or implied, including any implied warranties of merchantability, non-infringement of third party intellectual property rights, and fitness for a particular purpose.** Implementors of this Specification are advised to review the Liberty Alliance Project's website (http://www.projectliberty.org/) for information concerning any Necessary Claims Disclosure Notices that have been received by the Liberty Alliance Management Board.

Abstract:

Identity Services enable the infrastructure for identity-oriented services, thereby also enabling identity-based Web services. Identity-oriented services include the authentication and federation services delivered in Liberty Alliance Version 1.0 Specification. In Version 2.0 and later, identity services enable the Principals to effectively and efficiently direct businesses to share identity information (attributes) amongst them to deliver value-add services to the Principals.

Filename: liberty-phase-2-mrd-v1.0.pdf

Contents

1 Introduction

Identity Services enable the infrastructure for identity-oriented services, thereby also enabling identity-based Web services. Identity-oriented services include the authentication and federation services delivered in Liberty Alliance Version 1.0 Specification. In Version 2.0 and later, identity services enable the Principals to effectively and efficiently direct businesses to share identity information (attributes) amongst them to deliver value-add services to the Principals.

2 Requirements for Phase 2

2.1 Common Requirements (for Phase 2 only)

Req#	UC #	Requirements (for Phase/Version 2 only)
1	3.1.1, 3.1.2	Mechanism for an Attribute Provider (AP) or an Attribute Broker (AB) to register/deregister with the Discovery Service (DS) a list of attribute classes it provides for a specific Principal.
2	3.1.2	Mechanism for a DS to determine the type of entity (AB or AP) being registered or deregistered.
3	3.1.2	Mechanism for the DS to prompt the user during the registration process (e.g., to confirm the registration). Such mechanism(s) should support the ability to allow the requestor to prompt the user, asking the requestor to direct the user to the DS's site, or the DS using a LECP communications channel to ask the user directly.
4	3.1.3	Mechanism for an AB to discover the APs registered for an attribute class at the DS.
5	3.1.3	Mechanism for an AB to indicate to an AP that it is making the request on behalf of the originating SP (as opposed to the AB itself). This will enable the AP to make the correct authorization decision for the attribute access request.
6	3.1.4	Mechanism for an SP to direct Principal to an AP to set or modify value of one or more attributes, and be directed back to SP.
7	3.1.5	Mechanism for a provider to create or update Principal's attributes at an AP.
8	3.1.6	Mechanism for an AP to request consent from the Principal through the SP. This mechanism should protect itself from unauthorized use.
9	3.1.6	Mechanism for the SP to provide the results of that consent query to the AP.
10	3.1.7	Mechanism for an AP to request that the SP direct the Principal to the AP to request the Principal for consent.
11	3.1.8	Mechanism for an AP to utilize a LECP communications channel for querying the Principal's consent and obtaining the Principal's response.
12	3.1.8	Mechanism for an SP to indicate that the request is an authorization check request.
13	3.1.8	Mechanism to include the association of authorization context as part of the attribute request.

14	3.1.9	Mechanism for the IdP to include Principal's attributes in the authentication response/assertion to the SP. Note it is out of scope for Liberty to define what set of attributes the IdP includes in the authentication response.
15	3.1.9	Mechanism for an SP to request Principal's attributes in an authentication request to the IdP. Note it is out of scope for Liberty to define what set of attributes the IdP includes in the authentication response.
16	3.1.10	Guidelines for an AB to enable Principals to manage their attribute access permissions and usage directives at the AB for all the APs working with that AB.
17	3.1.11	Mechanism for an SP to indicate to the IdP that a federation is a federation to the affiliation and not to the SP itself.
18	3.1.11	Mechanism for an SP or IdP to request a list of members in an affiliation.
19	3.1.11	Mechanism for an SP to represent on an authentication request that they are acting as a member of an affiliation.
20	3.1.12	Mechanism for an SP to indicate “acting as” affiliation on attribute requests.
21	3.1.12	Mechanism for an AP to verify affiliation membership indicated on attribute requests.
22	3.1.13	Mechanism for an SP to make anonymous attribute requests and receive anonymous attribute responses. In other words, the ability to share attributes without disclosing the identity of the Principal to the requestor or SP.
23	3.1.14	Mechanism for an SP to associate intended usage with the corresponding requested attributes in an attribute request to an AP.
24	3.1.14	Mechanism for an AP to associate the agreed upon intended usage directives with the attribute response.
25	3.1.15	Mechanism for an AP to return a list of acceptable usage directives to an SP when the intended usage does not match the Principal's usage directives.
26	3.1.15	Guideline for APs (in the usage negotiation scenario) to reply, always, to an SP's attribute request with usage directives that, for privacy purposes, are equal to or stricter than those originally stated in the SP's attribute request.
27	General	Mechanism for an AP, an AB, or a DS to provide denial of access and reason for denial in response to attribute or discovery requests from other providers.
28	General	Mechanism for an AP or AB to partially fulfill requests for attributes.

29	General	Mechanism for an AP or an AB to associate Principal's consent for his/her permissions for an SP for a given set of attributes when the set of attributes are shared with the SP.
30	General	Mechanism for an AP or an AB to identify and authenticate the providers who make attribute requests.
31	General	Mechanism for an SP to request one or more individual attributes and/or a Class of Attributes in a single request for a specified Principal from an AP and/or AB.
32	General	Mechanism for an AP to send one or more attributes in a single response to the requesting entity.

2.2 Interactions Across Authentication Domains Requirements

33	3.2.1	Mechanism for an IdP (IdP2) to introduce, securely, an SP or AP in its authentication domain and an IdP (IdP1) in another authentication domain to each other on behalf of a Principal so that they can work with each other using standard Liberty protocols as if they were in the same authentication domain.
34	3.2.1	The "introducing" IdP (IdP2) must have the option of being able to track introductions for the Principals from IdP1 for which it has performed the introductions. Note that the IdP2 will only be able to identify the Principals using pseudonyms provided by IdP1, not local identities on IdP2. However, the same pseudonym would be used for the same Principal federating with multiple SP/APs at IdP2, so IdP2 would have a list of all of the SP/APs for a particular Principal.
35	3.2.1	Mechanism to recognize SP's authentication domain IdP in a federation request to Principal's IdP.
36	3.2.1	Mechanism for SP's authentication domain IdP to communicate Principal's current IdP in authentication response.
37	3.2.1	Mechanism for IdP2 to notify interested entities within its authentication domain of a severed relationship with another IdP.
38	3.2.3, 3.2.2, 3.2.4	Mechanism for SP to discover the chain of authentications.
39	3.2.5	Mechanism for SPs and IdPs to obtain, dynamically, each other's information, including necessary crypto/trust credentials.
40	3.2.5	Guidelines for establishing verifiable trust relationship using existing cryptographic and trust management technologies and business practices. Such mechanisms already exist and may be referenced by Liberty implementation guidelines.
41	3.3.2	Mechanism for an LECP to act on behalf of a Principal and perform actions as that Principal at an SP/IdP/AP.

2.3 Person Identity Profile Service Requirements

Req#	Requirements (for Phase/Version 2 only)
42	Mechanism to enable providers share the following core identity attributes in an interoperable way: a. Name of the Principal b. Billing Address c. Shipping Address d. Daytime Telephone Number e. Evening Telephone Number f. Mobile Telephone Number g. Fax Number h. Email Address i. Preferred Language j. Preferred Time Zone k. Date of Birth l. Gender m. Employer Info 1. Employer Name 2. Employer Address 3. Employer URI 4. Employee Id 5. Employer Telephone Number 6. Department n. Job Title **NOTE:** All attributes standardized for this profile are NOT required to be present at all times in every identity profile. Also, each of the above attributes can have zero or more values.
43	For each of the above identity attributes, the specification must specify the names and semantics of the attribute schemas to enable interoperability.
44	Mechanism for a provider to discover the Core Profile Service of a specified Principal.

3 Use Cases

3.1 Identity Services

Identity Services enable the infrastructure for identity-oriented services, thereby also enabling identity-based Web services. Identity-oriented services include the authentication and federation services delivered in Liberty Alliance Version 1.0 Specification. In Version 2.0 and later, identity services enable the Principals to effectively and efficiently direct businesses to share identity information (attributes) amongst them to deliver value-add services to the Principals.

3.1.1 AP/AB Notifies DS of an Updated List of Provided Attribute Classes or Termination of Service for Principal

Title/ID	AP/AB Notifies DS of an Updated List of Provided Attribute Classes or Termination of Service for a Principal
Pre-Conditions	1. Principal has an account with, and is authenticated by, IdP. 2. AP/AB is registered with the relevant DS.
Constituents	IdP, AP/AB, Principal
Use Case	1. AP/AB informs the relevant DS that it wishes to stop serving as Attribute Provider on behalf of Principal: either as a whole or only for a list of specific Attribute classes. 2. DS updates Principal's record with new list of Attribute classes that AP/AB handles.
Post Conditions	None
Alternate Courses of Action	
Requirements	1. Mechanism for an Attribute Provider (AP) or an Attribute Broker (AB) to register/deregister with the Discovery Service (DS) a list of attribute classes it provides for a specific Principal.

3.1.2 AP/AB Registers at DS

An AP/AB contacts DS to let it know that it is capable of handling attribute requests for specific attribute classes. We call this process "registration" of AP/AB with the DS. This process can take place on the initiative of the AP/AB. It can also take place dynamically during a transaction on the initiative of the Principal. This use case describes the former.

If a payment business model is used for releasing attributes, the entity will collect payment and manage all data flow for AP. This entity is called an Attribute Broker. This facilitates fewer bilateral business agreements between APs and SPs as SPs can enter into agreements with the AB, which in turn has agreements with the APs.

Title/ID	AP/AB Registers at DS
Pre-Conditions	1. Principal has an account at the IdP. 2. Principal has authenticated at the IdP.
Constituents	AP/AB, Principal, DS, IdP
Use Case	1. Principal is at AP/AB (acting as SP). 2. Principal consents to federation of AP/AB. 3. AP/AB federates with Principal's identity at the IdP. 4. AP/AB submits registration request at the DS for specified attribute class or classes. 5. DS processes registration request.
Post Conditions	AP/AB is registered at DS to service the specified attribute class or classes on behalf of Principal.
Alternate Courses of Action	5A. DS determines that the specified class or classes are already serviced by an AP/AB and requests confirmation from the Principal before registering the new AP/AB with the classes of attributes that are already registered with the DS.
Requirements	1. Mechanism for an Attribute Provider (AP) or an Attribute Broker (AB) to register/deregister with the Discovery Service (DS) a list of attribute classes it provides for a specific Principal. 2. Mechanism for a DS to determine the type of entity (AB or AP) being registered or deregistered. 3. Mechanism for the DS to prompt the user during the registration process (e.g., to confirm the registration). Such mechanism(s) should support the ability to allow the requestor to prompt the user, asking the requestor to direct the user to the DS's site, or the DS using a LECP communications channel to ask the user directly.

3.1.3 SP Attribute Request Resolved Through AB

Title/ID	SP Attribute Request Resolved Through AB.
Pre-Conditions	1. Principal has an account at the IdP. 2. Principal has authenticated at the IdP. 3. Principal has an AB for a particular class of attributes. 4. Principal has one or more APs for that class of attributes. 5. Principal has federated to an SP.
Constituents	AB, SP, Principal, DS, IdP
Use Case	1. Principal is at SP. 2, SP requests the AP(s) for a particular class of attributes from DS. 3. DS responds with AB information (which looks like AP to SP). 4. SP submits attribute request to AB. 5. AB requests contact info for APs from DS for the particular class of attributes. 6. DS responds with AP information for each of the attribute classes. 7. AB requests attributes from each of the APs, indicating that the original request is coming from the SP. 8. AP checks access controls and if permissions allow, processes the attribute request from the SP. 9. AP(s) return attributes to AB. 10. AB returns attributes to SP.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for an AB to discover the APs registered for an attribute class at the DS. 2. Mechanism for an AB to indicate to an AP that it is making the request on behalf of the originating SP (as opposed to the AB itself). This will enable the AP to make the correct authorization decision for the attribute access request.

3.1.4 SP Directs Principal to AP to Set Attribute Values

Title/ID	SP Directs Principal to AP to Set Attribute Values
Pre-Conditions	1. Principal has authenticated at an IdP. 2. Principal has federated its identity with SP and AP.
Constituents	SP, AP, Principal
Use Case	1. Principal is at SP. 2. SP requests from DS the Principal's AP for attribute classes. 3. DS directs SP to AB. 4. SP requests the attributes from AP. 5. AP determines that the requested attribute values (one or more) have not been previously set by the Principal. 6. AP responds to SP indicating that the requested attribute values are not set. 7. SP directs the Principal to AP to set the attribute values. 8. AP prompts Principal to enter values of Attributes and related permissions online and gathers the attribute values. 9. AP directs the Principal back to the SP. AP provides the SP with requested attribute values.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for an SP to direct Principal to an AP to set or modify value of one or more attributes, and be directed back to SP.

3.1.5 SP Prompts Principal for Missing Attribute Values, Then SP Updates Principal's Attribute Values at AP

This use case is important in certain e-commerce situations where the SP wants to keep the user at their site for the entire duration of a transaction without directing the user to the AP.

Title/ID	SP Prompts Principal for Missing Attribute Values, then SP Updates Principal's Attribute Values at AP
Pre-Conditions	1. Principal has an account with AP. 2. SP has requested attributes from AP (either through Principal's IdP or by asking Principal). 3. AP does not have value of attributes for Principal.
Constituents	SP, AP, Principal
Use Case	1. SP prompts Principal to enter value of attributes through front channel (directly through user agent). 2. Principal enters value of attributes. 3. After it has finished the transaction, SP submits value of attributes back to AP (update of attributes at the AP by the SP). 4. AP checks access controls and if permissions allow, processes the attribute update request from the SP.
Post Conditions	None.
Alternate Courses of Action	
Requirements	1. Mechanism for a provider to create or update Principal's attributes at an AP.

3.1.6 Access Mgmt: Exception Case – AP Trusts SP to Query Principal

Title/ID	Exception case – AP Trusts SP to Query Principal
Pre-Conditions	1. Principal has authenticated at IdP. 2. Principal has federated its identity with SP and AP. 3. Principal is at SP. 4. SP has assertion to talk to AP on behalf of Principal. 5. AP trusts SP to ask the Principal for consent.
Constituents	Principal, IdP, SP, AP
Use Case	1. SP requests an attribute from AP. 2. AP checks access controls and discovers that the permissions to that SP are indeterminate. 3. AP responds to SP indicating that the status is indeterminate and provides a consent query to present to the Principal. 4. The SP presents the query to the Principal. 5. Principal responds positively to the SP. 6. SP conveys response to AP. 7. SP resubmits request with the consent for the attribute. 8. AP records that the consent was obtained. 9. AP provides the requested attribute to the SP.
Post Conditions	
Alternate Courses of Action	Pre-authorization Case: Substitute #9 above--AP responds positively to the SP therefore allowing the SP to make the request in the future.
Requirements	1. Mechanism for an AP to request consent from the Principal through the SP. This mechanism should protect itself from unauthorized use. 2. Mechanism for the SP to provide the results of that consent query to the AP.

3.1.7 Access Mgmt: Exception Case – SP Directs Principal to AP

Title/ID	Exception Case – SP Directs Principal to AP.
Pre-Conditions	1. Principal has authenticated with IdP. 2. Principal has federated its identity with SP and AP. 3. Principal is at SP. 4. SP has an assertion from DS to talk to AP on behalf of Principal. 5. In this case, the AP does not trust the SP.
Constituents	Principal, IdP, SP, AP
Use Case	1. SP requests an attribute from AP. 2. AP checks their access controls and discovers that the permissions to that SP are indeterminate. 3. AP responds to SP indicating that the status is indeterminate and asks the SP to direct the Principal to the AP so that the AP can ask the Principal for consent. 4. AP presents the query to the Principal. 5. Principal responds positively to the AP. 6. AP records the consent. 7. AP directs the Principal to the SP. 8. SP resubmits the request.
Post Conditions	
Alternate Courses of Action	Pre-authorization Case: Substitute #5 above--AP responds positively to the Trusted Site therefore allowing the SP to make the request in the future.
Requirements	1. Mechanism for an AP to request that the SP direct the Principal to the AP to request the Principal for consent.

3.1.8 Access Mgmt: Exception case – AP Queries Consent from the Principal Through a Liberty-Enabled Client or Proxy (LECP)

Title/ID	Exception Case – AP Queries Consent from Principal Through a Liberty-Enabled Client or Proxy (LECP)
Pre-Conditions	1. Principal has authenticated with IdP. 2. IdP is authenticated through LECP that is maintaining a communications channel to Principal's identity. 3. Principal has federated its identity with SP and AP. 4. Principal is at SP. 5. SP has assertion to talk to AP on behalf of Principal.
Constituents	Principal, IdP, SP, AP
Use Case	1. SP requests an attribute from AP. 2. AP checks their access controls and discovers that the permissions to that SP are indeterminate. 3. AP requests consent from Principal through the communications channel to the LECP. 4. Principal responds positively to the AP. 5. AP records the consent. 6. AP provides requested attribute to the SP.
Post Conditions	
Alternate Courses of Action	Authorization Check Case: Substitute #6 above--AP responds positively to the SP therefore allowing the SP to make the request in the future.
Requirements	1. Mechanism for an AP to utilize a LECP communications channel for querying the Principal's consent and obtaining the Principal's response. 2. Mechanism for an SP to indicate that the request is an authorization check request. 3. Mechanism to include the association of authorization context as part of the attribute request.

3.1.9 Attributes Included in Authentication Requests and Authentication Responses/Assertions

While corporations wish to increase efficiency by providing services to users originating from their partners, customers, and suppliers, they do not wish to take on the burden of administering and maintaining lists of "foreign" users. An SP provides services (e.g., 401K, health insurance) to users originating from many different enterprises (IdPs) without requiring them to have an account at the SP. Instead, users are identified by a set of attribute values provided by an AP. The SP provides services based on the value of attributes communicated to it by the AP/IdP. It will also require a minimum set of attributes to be communicated from the AP/IdP before providing access to any services.

Title/ID	Attributes Included in Authentication Requests and Authentication Responses/Assertions
Pre-Conditions	1. Principal has an account with an IdP and an AP. In this case, the IdP and AP are the same entity. 2. Principal has federated its account at the AP with its account at the IdP. 3. IdP trusts and has business agreements with an SP for attribute sharing. These agreements include a description of the attributes to be sent to the SP by the IdP (using the AP). 4. Principal need not have an account with the SP. 5. Principal has given the IdP plus AP the permission to share its specific attributes without disclosing Principal's identity (anonymous attribute sharing or anonymous disclosure).
Constituents	IdP, SP, AP, Principal
Use Case	1. Principal authenticates at the IdP and receives assertion with authentication and attribute information from the AP. 2. Principal is redirected or navigates to SP. 3. Principal presents assertion with authentication and attribute information to SP. 4. SP authenticates Principal against the assertion and provides appropriate access to services based upon attribute values found in the assertion.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for the IdP to include Principal's attributes in the authentication response/assertion to the SP. Note it is out of scope for Liberty to define what set of attributes the IdP includes in the authentication response. 2. Mechanism for an SP to request Principal's attributes in an authentication request to the IdP. Note it is out of scope for

	Liberty to define what set of attributes the IdP includes in the authentication response.

3.1.10 Principal Reviews/Updates Permissions at an AB

The purpose of this use case is to generate the requirement for a Principal to be able to navigate to an AB and review and set all AP's policy for sharing Attributes. While providing convenience to the Principal, this, additionally, allows the Principal to check on how the authorization provided by an AB to SPs will unroll. This facilitates privacy controls to Principals.

Title/ID	Principal Reviews/Sets/Modifies Permissions at an AB
Pre-Conditions	1. Principal has accounts at AB, APs, and IdP and federates them. 2. Principal is authenticated at IdP. 3. Principal is at AB.
Constituents	IdP, AB, APs, Principal
Use Case	1. Principal reviews list of SP for which he has set up access rights to access each Class of Attributes. 2. AB retrieves permissions at APs as needed to display the relevant information to the Principal. 3. Principal modifies permissions for one or more SPs. 4. AB pushes final permissions to APs upon Principal's request.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Guidelines for an AB to enable Principals to manage their attribute access permissions and usage directives at the AB for all the APs working with that AB.

3.1.11 Federation with Affiliations

This describes the process of federating with an Affiliation of SPs, as opposed to a single SP.

Title/ID	Federation with Affiliations
Pre-Conditions	1. Principal has an account with an IdP. 2. SP1 and SP2 are members of the affiliation federation.
Constituents	IdP, SP1, SP2, AF, Principal
Use Case	1. Principal authenticates at the IdP. 2. Principal navigates to SP1. 3. SP1 asks Principal if they would like to federate with the members of affiliation federation. 4. Principal requests list of members of affiliation federation from IdP. 5. IdP provides list of members. 6. Principal authorizes federation with affiliation federation members. 7. IdP records federation. 8. Principal browses to SP2. 9. SP2 recognizes Principal due to Principal's federation with affiliation federation.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for an SP to indicate to the IdP that a federation is a federation to the affiliation and not to the SP itself. 2. Mechanism for an SP or IdP to request a list of members in an affiliation. 3. Mechanism for an SP to represent on an authentication request that they are acting as a member of an affiliation.

3.1.12 Authorization with Affiliations

This is the process of authorizing access to attributes based upon membership in an affiliation.

Title/ID	Authorization with Affiliations
Pre-Conditions	1. Principal has an account with an IdP. 2. SP1 and SP2 are members of affiliation AF1. 3. Principal has federated with SP1 and SP2 (either directly or to the affiliation that SP1 and SP2 are members of). 4. Principal has attributes managed by AP.
Constituents	IdP, AP, SP1, SP2, AF1, Principal
Use Case	1. Principal authenticates at the IdP. 2. Principal navigates to SP1. 3. SP1 recognizes Principal. 4. Principal requests action that causes SP1 to require an attribute. 5. SP1 acting as a member of affiliation federation, requests attribute from AP. 6. AP determines that Principal has not yet granted access to that attribute for SP1 or the affiliation federation. 7. AP requests permission to distribute attribute to affiliation federation from Principal. 8. Principal authorizes distribution to affiliation federation. 9. AP records authorization. 10. AP returns attribute to SP1. 11. Principal completes business at SP1 and later navigates to SP2. 12. SP2 recognizes Principal due to prior federation. 13. Principal initiates action that requires attribute from AP. 14. SP2 acting as a member of affiliation federation, requests an attribute from AP. 15. AP returns attribute to SP2 based upon Principal's prior authorization to affiliation federation.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for an SP to indicate "acting as" affiliation on attribute requests. 2. Mechanism for an AP to verify affiliation membership indicated on attribute requests.

3.1.13 Anonymous Identity Services: Providing Services Without Sharing the Identity of Principal

An SP may wish to provide basic personalization services to its visitors/customers without requiring them to have an account at the SP or even identify themselves at the SP. Hence a user may *anonymously* share certain attributes with such SPs. For example, an SP may not require any sign-in by the user on their initial visits. Nor does the SP require the user to have an account at the SP. Yet, the SP may want to provide basic personalization based on attributes such as preferred language, gender, geo-location, time zone, etc. The user, when visiting the SP, may see content personalized to the user's preferences. Such personalization is the value-add provided by such SPs to attract customers and increase airtime and online time usage.

Note the anonymity is intended to protect the identity of the user. The SP never gets an identifier for the user, not even a repeatable pseudonym. Therefore, even if the user re-visits the SP a minute later, the SP would not know if it is the previous user visiting again. However, such anonymity (privacy) does not exist if the user willingly gives the permission to anonymously share his/her PII (Personally Identifiable Information) such as social security numbers, driver's licenses, passport numbers, etc.

Title/ID	Anonymous Identity Services: Providing Services Without Sharing the Identity of Principal
Pre-Conditions	1. Principal has an account with an IdP and an AP and has federated them. 2. IdP trusts and has business agreements with an SP for anonymous attribute sharing. 3. Principal need not have an account with the SP. 4. Principal has given the AP the permission to share its specific attributes without disclosing Principal's identity (anonymous attribute sharing or anonymous disclosure).
Constituents	IdP, SP, AP, Principal
Use Case	1. Principal authenticates at the IdP. 2. Principal navigates to SP. 3. SP verifies Principal's anonymous authentication assertion at IdP. 4. SP requests from DS the AP of desired attribute class(es) for the anonymous Principal. 5. DS responds to the SP with the contact info of AP, along with a confidential service ticket (readable only by the DS and the AP). 6. SP presents the confidential service ticket to the AP and requests the attributes from AP. 7. AP checks Principal's permissions and sends Principal’s attributes to SP without disclosing the identity of the Principal.
Post Conditions	

Alternate Courses of Action	
Requirements	1. Mechanism for an SP to make anonymous attribute requests and receive anonymous attribute responses. In other words, the ability to share attributes without disclosing the identity of the Principal to the requestor or SP.

3.1.14 Usage Directives: Attribute Requests that Include Intended Usage

Title/ID	Usage Directives: Attribute Requests that Include Intended Usage
Pre-Conditions	1. Principal has authenticated with IdP. 2. Principal has federated its identity with SP and AP. 3. Principal is at SP. 4. SP has assertion to talk to AP on behalf of Principal.
Constituents	Principal, IdP, SP, AP
Use Case	1. SP requests an attribute from AP and specifies intended usage. 2. AP checks access controls and determines that access is allowed. 3. AP checks that the intended use doesn't violate Principal's usage directives. 4. AP responds to SP with requested attributes, optionally associating the agreed upon usage directives (which will be the SP's stated intended usage).
Post Conditions	
Alternate Courses of Action	1. AP checks the intended usage and determines that it violates the Principal's usage directives. 2. AP responds to SP with failure response. 3. SP re-submits request with alternative intended usage (and starts the process over again).
Requirements	1. Mechanism for an SP to associate intended usage with the corresponding requested attributes in an attribute request to an AP. 2. Mechanism for an AP to associate the agreed upon intended usage directives with the attribute response.

3.1.15 Usage Directives: Negotiation of Usage Directives

Title/ID	Usage Directives: Negotiation of Usage Directives
Pre-Conditions	1. Principal has authenticated with IdP. 2. Principal has federated its identity with SP and AP. 3. Principal is at SP. 4. SP has assertion to talk to AP on behalf of Principal.
Constituents	Principal, IdP, SP, AP
Use Case	1. SP makes an attribute request to AP that includes the intended usage. 2. AP checks access controls and determines that access is allowed. 3. AP validates the intended usage against the Principal's usage directives and determines that the intended usage is not acceptable and therefore it cannot share the requested attribute. 4. AP denies access and responds with a list of acceptable usage directives that are more restrictive than the intended usage as stated by the SP. 5. SP resubmits the attribute request with new set of intended usages. 6. AP validates that the intended use doesn't violate its usage directives. 7. AP shares the attributes with the SP.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for an AP to return a list of acceptable usage directives to an SP when the intended usage does not match the Principal's usage directives. 2. Guideline for APs (in the usage negotiation scenario) to reply, always, to an SP's attribute request with usage directives that, for privacy purposes, are equal to or stricter than those originally stated in the SP's attribute request.

3.2 Interactions Across Authentication Domains

3.2.1 IdP Introduces SP in Its Domain to Another IdP in a Different Domain

To enable interoperability and interactions between authentication domains, an IdP in one domain needs to be able to introduce an SP in its domain to an IdP in a different domain.

Title/ID	IdP Introduces SP in Its Domain to Another IdP in a Different Domain
Pre-Conditions	1. IdP1 is a member of authentication domain 1. 2. Principal has an account with IdP1. 3. IdP2 and SP2 are members of authentication domain 2. 4. Principal has no account with IdP2. 5. Principal has an account with SP2. 6. IdP2 joins authentication domain 1. This is a key pre-condition because otherwise phase 1 specs do not allow authentication assertions to be shared or visible outside domain 1.
Constituents	IdP1, IdP2, SP2, Principal
Use Case	1. Principal authenticates at IdP1. 2. Principal navigates to SP2. 3. SP2, as per phase 1 specs, will not see any authentication assertions for the Principal. Hence, SP2 will direct the Principal to IdP2 for authentication. 4. IdP2, by virtue of being in authentication domain 1 and 2, has access to and recognizes the authentication assertions from IdP1. 5. IdP2 responds to SP2 with location of IdP1. 6. SP2 sends federation request to IdP1. 7. IdP1 requests IdP2 to validate SP2. 8. IdP2 validates SP2. 9. IdP1 completes the federation request with SP2 for the Principal.
Post Conditions	Principal's identity/account at IdP1 is now federated with that at the SP2.
Alternate Courses of Action	
Requirements	1. Mechanism for an IdP (IdP2) to introduce, securely, an SP or AP in its authentication domain and an IdP (IdP1) in another authentication domain to each other on behalf of a Principal so that they can work with each other using standard Liberty protocols as if they were in the same authentication domain. 2. The "introducing" IdP (IdP2) must have the option of being able to track introductions for the Principals from IdP1 for which it has performed the introductions. Note that the IdP2

	will only be able to identify the Principals using pseudonyms provided by IdP1, not local identities on IdP2. However, the same pseudonym would be used for the same Principal federating with multiple SP/APs at IdP2, so IdP2 would have a list of all of the SP/APs for a particular Principal. 3. Mechanism to recognize SP's authentication domain IdP in a federation request to Principal's IdP. 4. Mechanism for SP's authentication domain IdP to communicate Principal's current IdP in authentication response. 5. Mechanism for IdP2 to notify interested entities within its authentication domain of a severed relationship with another IdP.

3.2.2 Federated SSO/Authentication Across Two or More Authentication Domains

Enable SSO and authentication sharing across authentication domains where identity providers in each domain act as authentication brokers for other domains.

Title/ID	Federated SSO/Authentication Across Two or More Authentication Domains
Pre-Conditions	1. IdP1 is a member of authentication domain 1. 2. Principal has an account with IdP1. 3. IdP2 and SP2 are members of authentication domain 2. 4. Principal has no account with SP2 or IdP2. 5. IdP2 joins authentication domain 1. This is a key pre-condition because otherwise phase 1 specs do not allow authentication assertions to be shared or visible outside domain 1. 6. AB2 is a member of domains 1 and 2, and acts as a broker in domain 2.
Constituents	IdP1, IdP2, SP2, AB2, Principal
Use Case	1. Principal authenticates at IdP1. 2. Principal navigates to SP2. 3. SP2, as per phase 1 specs, will not see any authentication assertions for the Principal. Hence, SP2 will direct the Principal to IdP2 for authentication. 4. IdP2, by virtue of being in authentication domain 1 and 2, has access to and recognizes the authentication assertions from IdP1. 5. IdP2 creates virtual account for the Principal and flags that all attribute requests for the Principal are to be routed through AB2. 6. IdP2 directs the Principal back to SP2. 7. SP2 now sees the authentication assertion from IdP2 and lets the Principal access SP2 resources/services.
Post Conditions	
Alternate Courses of Action	1. If the user doesn't have an account with SP2, SP2 may deny the user access to SP2 resources/services.
Requirements	1. Mechanism for SP to discover the chain of authentications.

3.2.3 Identity Services Across Two Federated Authentication Domains through an AB

Title/ID	Identity services across two federated authentication domains through an AB.
Pre-Conditions	1. IdP1 and AP1 are members of authentication domain 1. 2. Principal has an account with IdP1. 3. IdP2 and SP2 are members of authentication domain 2. 4. Principal has no account with SP2 or IdP2. 5. IdP2 joins authentication domain 1. This is a key pre-condition because otherwise phase 1 specs don't allow authentication assertions to be shared or visible outside domain 1. 6. Principal has been authenticated by IdP2 as in 3.2.2. 7. AB2 is a member of domains 1 and 2 and acts as a broker in domain 2.
Constituents	IdP1, IdP2, SP2, AP1, DS1, DS2, AB2, Principal
Use Case	1. SP2 requests from DS2 the Principal's AP for attribute classes. 2. DS2 directs SP2 to AB2. 3. SP2 requests the attributes from AB2. 4. AB2, recognizing that the Principal is from domain 1, requests from DS1 the Principal's AP for the requested attribute classes. 5. DS1 directs AB2 to AP1. 6. AB2 requests the attributes from AP1. 7. AP1 responds to AB2 with the attributes. 8. AB2 responds to SP2 with the attributes.
Post Conditions	
Alternate Courses of Action	
Requirements	1. Mechanism for SP to discover the chain of authentications.

3.2.4 Multi-IdP Chained Principal Authentication

Title/ID	Multi-IdP Chained Principal Authentication
Pre-Conditions	1. Principal has an account at IdP1 and IdP2. 2. IdP1 has a business agreement with IdP2 by which IdP2 can reuse IdP1's Principal authentication sessions and statements. 3. SP does not have a contractual agreement with IdP1. 4. SP does have a contractual agreement with IdP2. 5. IdP1 authenticates Principal. 6. Principal has federated their accounts at IdP1 and IdP2 (acting as an SP).
Constituents	IdP1, IdP2, SP, Principal
Use Case	1. Principal request service from SP. 2. SP requires Principal authentication. 3. SP contacts IdP2 requesting authentication 4. IdP2 recognizes user has authentication with IdP1 5. IdP2 contacts IdP1 and request's Principal authentication. 6. IdP1 sends authentication information to IdP2. 7. IdP2 provides authentication assertions of the Principal to the SP.
Post Conditions	1. Principal can access the controlled services of SP
Alternate Courses of Action	
Requirements	1. Mechanism for SP to discover the chain of authentications.

3.2.5 SP Dynamically Joining the Authentication Domain of the IdP

Title/ID	SP Dynamically Joining the Authentication Domain of the IdP
Pre-Conditions	1. SP and IdP belong to different authentication domains. 2. Principal has accounts at both SP and IdP. 3. Principal wants to federate his SP/IdP accounts to facilitate SSO.
Constituents	Principal, IdP, SP
Use Case	1. Principal requests service from SP. 2. Principal provides name of the IdP to SP. 3. SP and IdP obtain each other's information including necessary crypto/trust credentials (various methods can be used for such discovery and exchange). 4. SP and IdP validate each other's credentials (e.g., both the SP and IdP develop and validate a certification path between their local trust anchor and each other's public-key certificate). 5. SP and IdP federate Principal's accounts and proceed with authentication.
Post Conditions	SP and IdP obtain and validate each other's information including necessary crypto/trust credentials enabling trust for federation of Principal's accounts, authentication, and SSO.
Alternate Courses of Action	In addition to using Principal's actions for triggering dynamic trust establishment, other stimuli can be used to facilitate the same scenario.
Requirements	1. Mechanism for SPs and IdPs to obtain, dynamically, each other's information, including necessary crypto/trust credentials. 2. Guidelines for establishing verifiable trust relationship using existing cryptographic and trust management technologies and business practices. Such mechanisms already exist and may be referenced by Liberty implementation guidelines.

3.3 Delegation of Authority

Enterprises and business need to automate their business operations/service tasks on behalf of their customers, employees, and/or partners. To enable this, businesses need delegated authority to act as the Principal's agent to access and update the Principal's accounts and/or attributes, while preserving the right of the Principal to authenticate/authorize at the time of actual service transaction. Delegation for such activities enables federation and Web services operations on behalf of the Principal and do not require the Principal to be involved during the actual transaction. Delegation and consent are provided separately from the actual transaction.

3.3.1 Delegation of Authority to Federate/Link Accounts

One or more Principals delegate authority to federate/link their accounts without requiring authentication of Principals at the time of linking.

In the enterprise and the B2B space, IT managers want to federate/link various accounts of each of their employees/suppliers/partners based on the employee/partner/supplier contracts when they joined the B2B/enterprise. In such markets, enterprise/B2B practices don't actually contact each employee/supplier/partner at the actual time of linking/federations; instead linking is done separately from the process of capturing consent of the Principals. Such business entities also have practices, for operational efficiency reasons, to enable such linking in bulk for all Principals.

Title/ID	Delegation of Authority to Federate/Link Accounts of a Principal
Pre-Conditions	1. IdP has an authenticated identity. 2. SP has an authenticated identity. 3. IdP has implemented a Liberty-compliant architecture to create, store, and manage identities. 4. One or more Principals have consented to the IdP's policies and delegated the authority to the IdP to link some or all of the Principal accounts at the SP without requiring the IdP to further solicit Principal's permission for each of those link establishments.
Constituents	IdP, SP, one or more Principals
Use Case	1. IdP, at its convenience, will invoke a Web service on behalf of the Principal to initiate the linking of Principal's account with the IdP with the specified accounts at the SP. The Web service implements Liberty protocols and schemas to automatically (i.e., without human intervention) link the specified accounts. This Web service will have a counterpart Web service at the SP end to complete the automatic linking of accounts. 2. The Web service at either end (SP and IdP) are acting as 'agents' of the Principals. Before they perform any linking, Web services at either end must authenticate each other's identity. 3. After successful agent authentication, the Web services proceed to link the specified accounts on behalf of the Principal.

	4. The SP and IdP acknowledge the completion of federation and linking.
Post Conditions	
Alternate Courses of Action	In the event of failed federation of one or more accounts, SP and IdP communicate the list of accounts for which federation failed and the reasons for failure of each of those federations.
Requirements	

3.3.2 Principal Delegating Authority to LECP to Act on Its Behalf with SP/IdP/AP

Principal delegates authority or authorizes a LECP to perform operations on a Principal's subscribed services provided by another SP. One example for this use case is the implementation in the mobile network. There are some Principals with limited capability, for example handsets and thin-clients, that are able to perform end-to-end SSL, but are not flexible enough to understand every single Liberty protocol. They use a proxy to communicate with Liberty-enabled entities. In this case the Principals will sometimes present their identities (i.e., X.509 cert or any digitally-signed assertion) by themselves through the SSL channel and sometimes present their identities by proxy server (delegated authority). In order for SPs to recognize that the two Principals are identical, the proxy server can act as a Principal that can add the same authentication assertion in both events.

Title/ID	Principal Delegating Authority to LECP to Act on Its Behalf with SP/IdP/AP
Pre-Conditions	1. Principal has a proxy service (PXY). 2. Principal has account at SP. 3. Principal has an account at AP. 4. Principal has federated their identity to their accounts at SP and AP. 5. Principal has delegated authority to PXY to perform operations on their behalf.
Constituents	SP, PXY, AP, Principal, IdP
Use Case	1. Principal authenticates to IdP via PXY. 2. Principal initiates action at SP through PXY. 3. SP obtains Principal’s identity through PXY. 4. SP needs service of AP to complete action. 5. SP Requests AP identification and contact info from DS. 6. SP contacts AP to request attributes. 7. AP verifies access permissions and returns attributes to SP. 8. SP completes action returning.
Post Conditions	
Alternate Courses of Action	1. Principal attempts to perform same action at SP through direct SSL channel (which cannot be “proxied”).
Requirements	1. Mechanism for an LECP to act on behalf of a Principal and perform actions as that Principal at an SP/IdP/AP.

4 New Glossary Terms

4.1 Affiliation

A group of SPs organized to act as a single entity from the point of view of the customer. This is usually due to the group acting as a portal or acting as a single company.

4.2 Attribute Broker (AB)

- Serves as a relay for receiving attribute requests and sending attribute responses on behalf of multiple APs. The AB adheres to the business policies of the APs and permissions of the Principals.
- Facilitating fewer business relationships and agreements between entities, APs can benefit from one Attribute Broker signing up with several SPs on their behalf.
- From the perspective of an SP making an attribute request, AB and AP have the same interface.
- Unlike APs, ABs can interact with other APs.
- For the purpose of Version 2.0 prioritization, the following are restrictions being placed on the role of ABs for simplicity.
  - Only one AB may exist for a class of attributes per Principal.
  - If an AB exists for a class of attributes for a Principal, all APs who wish to provide service to the Principal using that class of attributes must work through the AB.
  - NOTE: Even with the above restrictions, MRD2 allows different Attribute Brokers for different classes for the same Principal. Furthermore, the phase 2 MRD also allows different Attribute Brokers for the same attribute class for different Principals.
  - In future phases, Liberty may relax the above restrictions based on business justifications.

4.3 Attribute Class

A predefined set of attributes such as the constituents of a Principal's name (prefix, first name, middle name, last name, and suffix). Liberty entities may standardize such classes. Some Liberty entities may also define classes that are used only within their business communities and not Liberty-wide.

NOTE:

Unless otherwise explicitly stated, the term "attribute" is used to denote "attribute names" in attribute requests and "attribute names and attribute values" in attribute responses.

4.4 Attribute Provider (AP)

- Provides attributes to a requester according to its own policies and Principal's Permissions.
- May delegate the actual hosting of the attributes to another entity. This would be transparent and irrelevant to IdPs/SPs involved in transactions, thus out of scope of Liberty.

4.5 Discovery Service (DS)

Has the ability to direct attribute requestors to the relevant AP or AB who provides requested classes of attributes for the specified Principal. For this market requirement document, member companies have not stated an explicit and immediate business need for the specification of a protocol between DS and IdP. However, such specifications may be considered in future versions as a business need arises.

4.6 Permissions

For the purpose of this document, the term 'permissions' encompasses both access controls and usage directives, unless otherwise explicitly stated.

4.7 Web Services

A Web service is a software application, identified by a URI, whose interfaces and binding are capable of being defined, described, and discovered by XML artifacts, and supports direct interactions with other software applications using XML-based messages such as SOAP via Internet-based protocols.