White Paper on Liberty Integration

Perspective of Integration of PayCircle into ID-WSF as an SiS

AA-004-001

Authors:
Fulup Ar Foll, Sun Microsystems
Bertolt Eicke, Siemens
Karsten Lüttge, Siemens

Table of Contents

Executive Summary

- Mobile identity management and 3rd party payment should and can be integrated
- Liberty mechanisms may be used for authentification, user confirmation and privacy enforcement
- PayCircle mechanisms may be used for payment transactions
- This applies also for the more general case, where subscriber and merchant belong to different payment service providers/circles of trusts, also known as the 4-box model.
- Slight modifications/extensions are needed in PayCircle WSI and possibly Liberty.
- PayCircle could be extended to be Liberty compliant, which could be even done without any formal agreement between the fora.
- Additionally PayCircle could even be the basis for an upcoming Liberty Payment SiS. Contractual issues are outside the sope of this document.
- The document may serve as the basis for a proof-of-concept including a joint reference implementation and a joint demonstrator.

1 Introduction

1.1 Context

This work was initiated by the PayCircle-Liberty liaison.

1.2 Audience

This document addresses technical staff and management interested in the integration of 3rd party payment with identity management.

The document should serve as discussion input to be communicated in PayCircle and Liberty.

1.3 Goals

PayCircle has since long identified the need to integrate payment with mechanisms for identity management, user confirmation and privacy enforcement. These requirements are not specific to payment but apply to the other web service interfaces in the Parlay X suite as well.

Liberty phase 3 will specifiy specific services (SiS) based on the general framework of phase 2. Requirement drafts for a Payment SiS as well as a Wallet SiS are already available.

The goal of the investigation is to explore an integration of identity management and payment based on (possibly extended) specifications of the Liberty Alliance Project and PayCircle.mailto:charge@once

2 Technical Concept

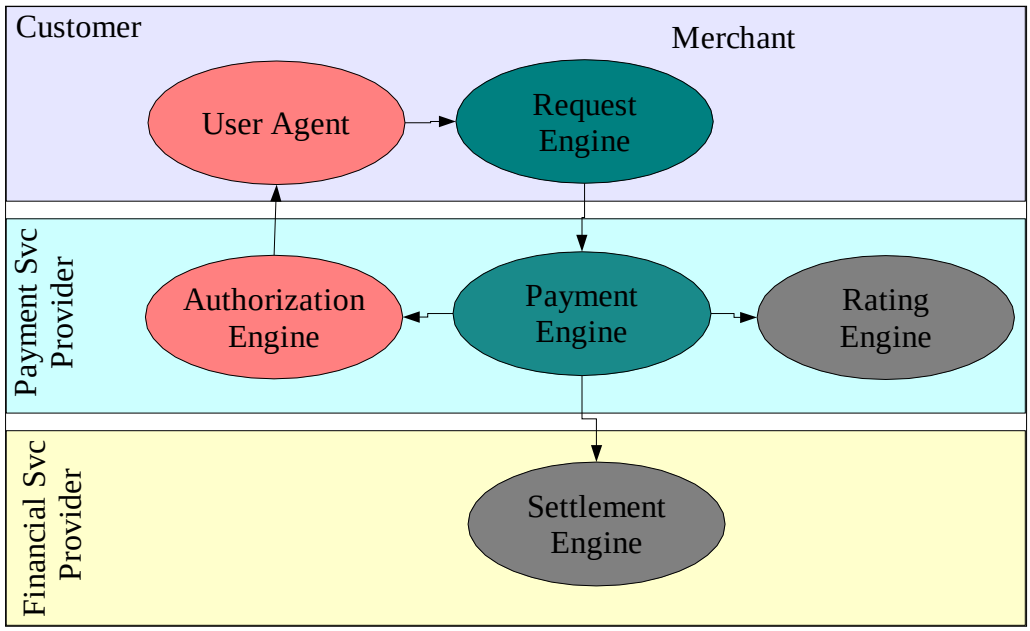
2.1 PayCircle

PayCircle® is a vendor-independent non-profit organization. Its main focus is to accelerate the use of payment technology and develop or adopt open payment APIs (uniform Application Programming Interfaces) based on XML, SOAP, Java and other Internet languages.

PayCircle deliverables include the Payment Web Service Specification which is identical and co-branded with the Payment part of the Parlay X specification. Additional PayCircle deliverables are white papers, use cases and reference implementation and sample clients for the specification.

PayCircle also endorses the JPay effort (JSR 182) to harmonize payment interfaces for the WSDL and Java communities.

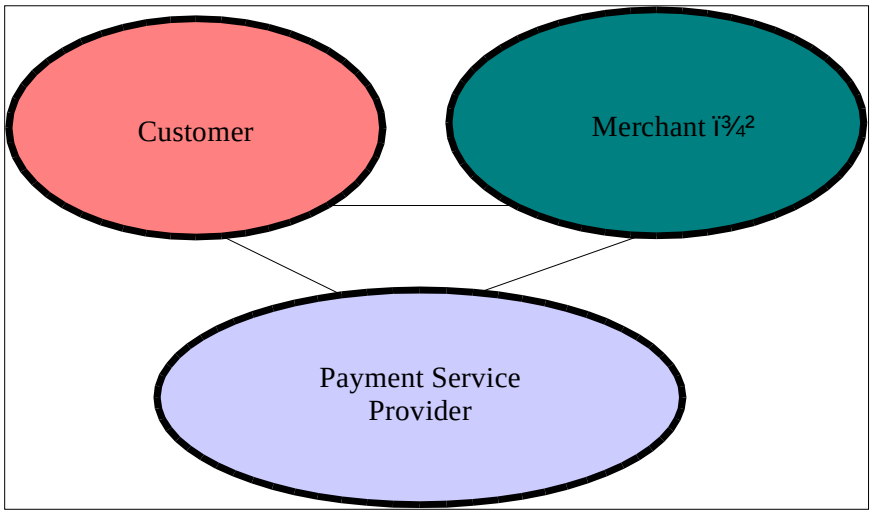
The PayCircle functional model is given by the following figure:



The entities shown belong to the involved business roles, the most important being the

- Customer (also termed end user or subscriber)
- Merchant (also termed 3rd party service provider)
- Payment Service Provider (PSP)

This gives rise to the 3-box architectural model



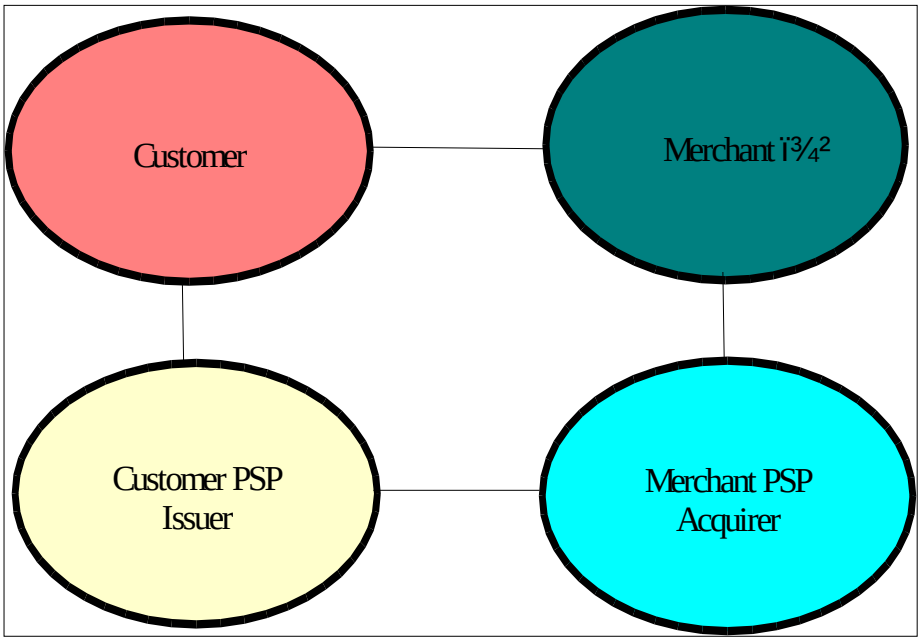
In this model the Payment WSI is only covering the Merchant-PSP interface.

In the more general case the merchant and the subscriber have business relationships with two distinct PSPs. Thus the subroles

- Issuer (Customer PSP)
- Acquirer (Merchant PSP)

are introduced which gives rise to the 4-box model which is also used as the OMA reference model for m-commerce:

This gives rise to the 4-box architectural model



In this model the Payment WSI is only covering the Merchant-Acquirer interface.

The 4-box model is generalizing the typical credit card model to arbitrary payment service providers, like mobile operators. It applies each time a customer is willing to buy some good from a merchant that while not having business relationship with this customer's payment service provider nevertheless has a business relationship with someone else that is trusted by his payment service provider. This scenario may happen in roaming cases, e.g. where an Orange customer while traveling in Germany is willing to buy some goods from a merchant that only have a relationship with T-Mobile. Note however that this interworking of payment service providers is independent of the physical bearer roaming. Thus a travelling user may use 3rd party services from his home PSP and a user may use foreign PSP services even when staying within his home network.

Special solutions may be deployed if there is a special trust relationship between the involved operators, e.g. between different branches of the same operator (e.g. Orange France and UK, Vodafone UK & Germany, ...).

This model may be refined by introducing an intermediate (broker) between the two payment service providers, an approach which is currently followed by a European operator consortium.

2.2 Liberty

Liberty Alliance was formed in December 2001 to serve as the premier open Alliance for federated network identity management and services. Its goals are to ensure interoperability, support privacy and promote adoption of its specifications, guidelines and best practices. As an industry consortium, it

is working to define standards in the identity management space. The Alliance has grown from just under 20 companies to more than 160 companies representing a worldwide cross-section of organizations, ranging from educational institutions and government organizations, to service providers and financial institutions, to technology firms and wireless providers.

The Liberty Alliance is developing and delivering specifications in a phased approach that allows for quicker and easier implementation of identity-based solutions. The Liberty Alliance released its first set of open specifications for federated identity management, in July 2002 and since then follow and extremely tight time line to upgrade releases.

Phase 1 enables federated identity management. It provides standards for single sign on and linking of disparate accounts within an affiliated group. With phase 1, a business can allow their users to sign in to an existing account once with a member of an affiliated group and navigate to various sites among the group without signing on again. Phase 1 specifications provides the plumbing for federated identity management. This body of work is referred to as the Liberty Alliance's Identity Federation Framework (ID-FF). Main features are:

- Opt-in account linking
- Single sign-on (SSO)
- Single logout
- Pseudonimity

Phase 2 enhances identity federation framework and enables interoperable identity-based Web services. While phase-2 is mainly focusing on Web services framework, it nevertheless introduces some updates in phase-1 federation framework in order to bootstrap Web services and to provide two phase-3 services (discovery service & personal profile). This new body of work is referred to as the Liberty Alliance's Identity Web Services Framework (ID-WSF). Main features are:

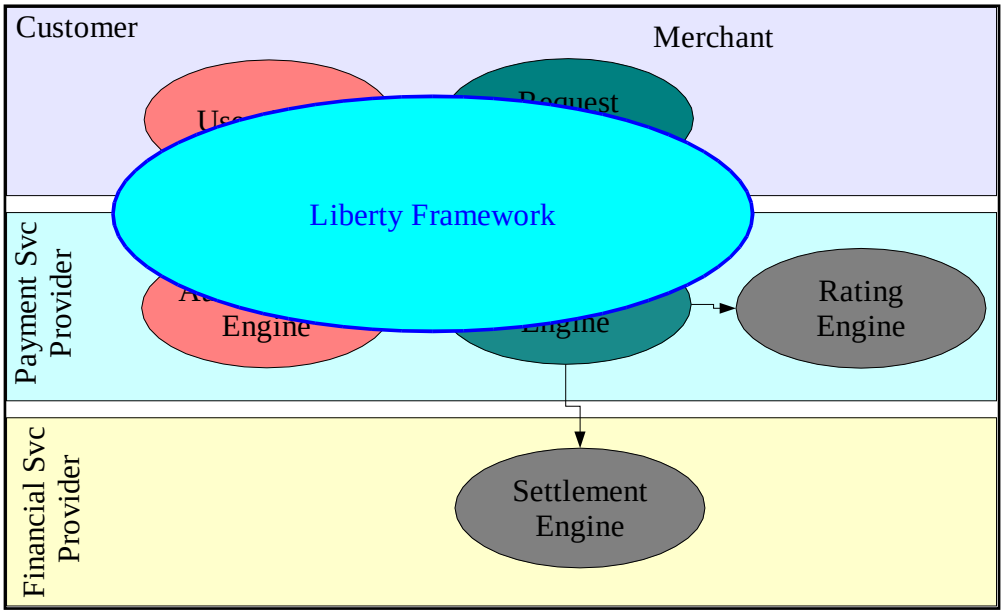
- Identity data services definition
- Identity based service discovery & invocation
- Permission based attribute sharing
- Interaction services
- Anonymity

Phase 3 and beyond, will develop specification of interoperable personalized services based on ID-WSF framework. This new body of work is referred to as the Liberty Alliance's Service Interface Specifications (ID-SIS), services are still in definition some, the first one under definition are:

- Contact book
- Geo. location
- Presence
- Calendar, etc.

2.3 Liberty Area of Influence on PayCircle

As explained previously Liberty is providing a full webservice framework that can leverage user identity while keeping control onto privacy & security issues for all participants: customers, identity provider and service provider. As Liberty does only normalize the relationship in between the different participants, its impact is not on the PayCircle logic but on how this logic interact with the others.



3 PayCircle as a Liberty SiS

The integration of PayCircle as a Liberty SiS should be quite straight forward. The highest level of complexity is arising with the 4 box model, where Liberty can allow a customer to make business with a merchant that has no direct business agreement with his identity provider.

While merchant and more generally selling/buying mechanism is very well understood, the implementation of such application requires many advanced features, that Liberty framework provides. The basic logic is very simple, the customer is willing to buy, the merchant is willing to sell and payment service provider is more than happy to guaranty the transaction for a small percentage of the value.

Nevertheless the actors involved have contradictory interests:

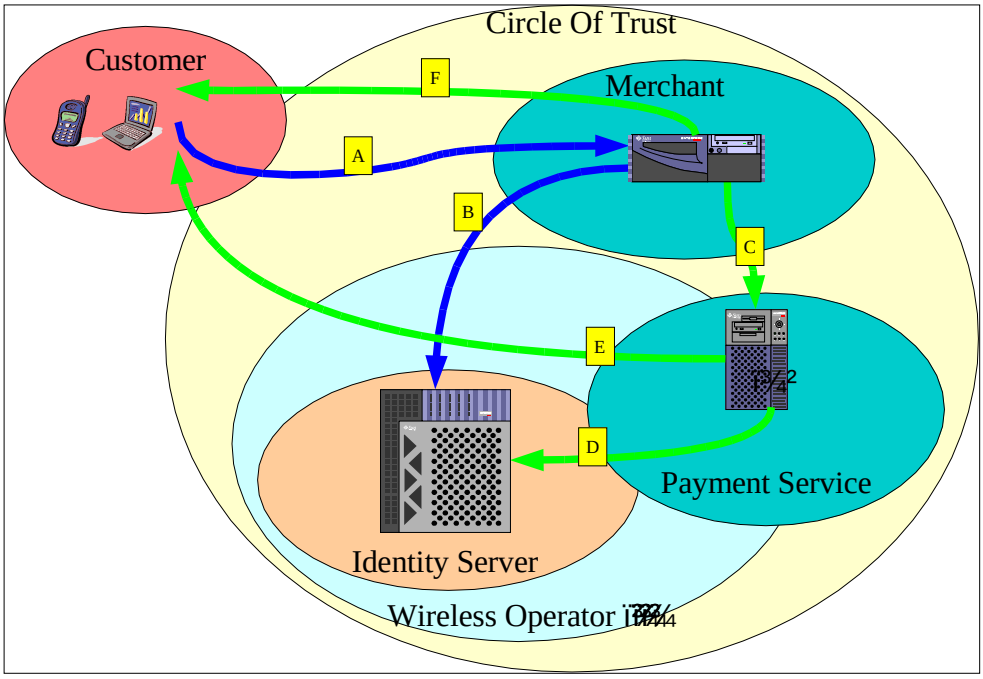
- Customer is willing to have the option to buy from any merchant, but is reluctant to trust most of them.
- Merchant is willing to sell to anyone, making customer experience seamless « one click buy », while assuring that he will eventually be paid.
- Payment service provider is willing to leverage the direct business relationship they have with their customers, while assuring that opening their user base will not lead them to customer leak.

Furthermore, business relationship in between the actors is limited and not easy to change.

- Customer only has a commercial contract with his payment service provider, e.g. wireless operator
- Merchant may sign an agreement with a couple of payment services, but not with hundreds. Credit card is a good model to envision how big this number could be.
- Payment service, might at least for micro payment be handled directly by the payment service provider, or be delegated to a trusted partner, example ”Orange UK” trusting “Orange France” for roaming users.

3.1 3-Box Model

The following image describes a possible scenario for payment using Liberty framework. In the decription below a wireless operator is acting as identity service and payment service, the same flow applies also for different actors. The main value from Liberty in this case is that customer does not have to trust the merchant, the operator does not have to reveal its customer name or phone number, payment service can either be an embedded service from the operator or an external trusted partner, in this case it knowlege of the customer will be limited to a pseudonym.



Simplified sequence flow

A) Customer is browsing merchant site and sends a purchase request. At this point the merchant site may create a browsing session for the user, but this one is still not authenticated.

B) Merchant needs to authenticate customer, it sends authentication request to the identity server (note: *Liberty proposed mechanisms to discover IDP are not described here)*. It receives in return an SSO (Single Sign On) authentication respond token. (*Note that the SSO token leverages Liberty ID-FF Federation mechanisms and thus hides real customer identity).*

C) Merchant is willing to request payment service to guaranty its transaction. First merchant site should discover customer payment service, using the ID-WSF bootstrap information it gets with the authentication response, then it requests the payment service on behalf of the

principal. This step leverages the fact that Liberty discovery mechanism is oriented « per principal », allowing merchant to request payment while not knowing the real customer identity. This step actually may include several interactions related to a payment transaction like reservation and subsequent final capture.

D) Payment Service connects onto operator charging service to secure its transaction. In order to do so, payment service like merchant previously, discovers operator charging service through Liberty DS, in order to get a token that will allow it to request charging on behalf of the principal.

E) User Interaction, before charging the customer, the payment service needs to get formal user consent. In simple cases consent might be given globally for a set of services within customer profile, but for buying action it is most likely that users will prefer to be interactively requested for consent. Liberty proposes three models for getting user consent, in our scenario the three could be valid and the choice will depend on business agreement, customer experience, ... Depending on the chosen model, consent can either:

- Go through merchant site, using it as a proxy for consent request, this model imposes to trust the merchant, or to force customer to sign the answer. This is valid only if there is a physical connection between the customer and the merchant, e.g. In Web or WAP sessions.
- Redirecting user onto the payment service, this is probably the most typical option for web application, it is especially valid if payment service is handled by a very well known and trusted entity [ex: a well known bank, a national wireless operator, ...]. This is valid only if there is a physical connection between the customer and the merchant, e.g. In Web or WAP sessions.
- Use an independent interaction service, in this case payment service discover the principal « Interaction Service » requesting the wireless operator DS. This one can then use any technique to get user consent « SMS, WAP Push, ... » This last option allow « out of band » consent request, which is mandatory in the case someone is willing to something in your name while you're not the one doing the request [ex: are you accepting to recharge your pre-paid account ?]. This is valid also in scenarios where there is no physical connection between the customer and the merchant. However, in browsing sessions, it will not yield the preferred customer experience.

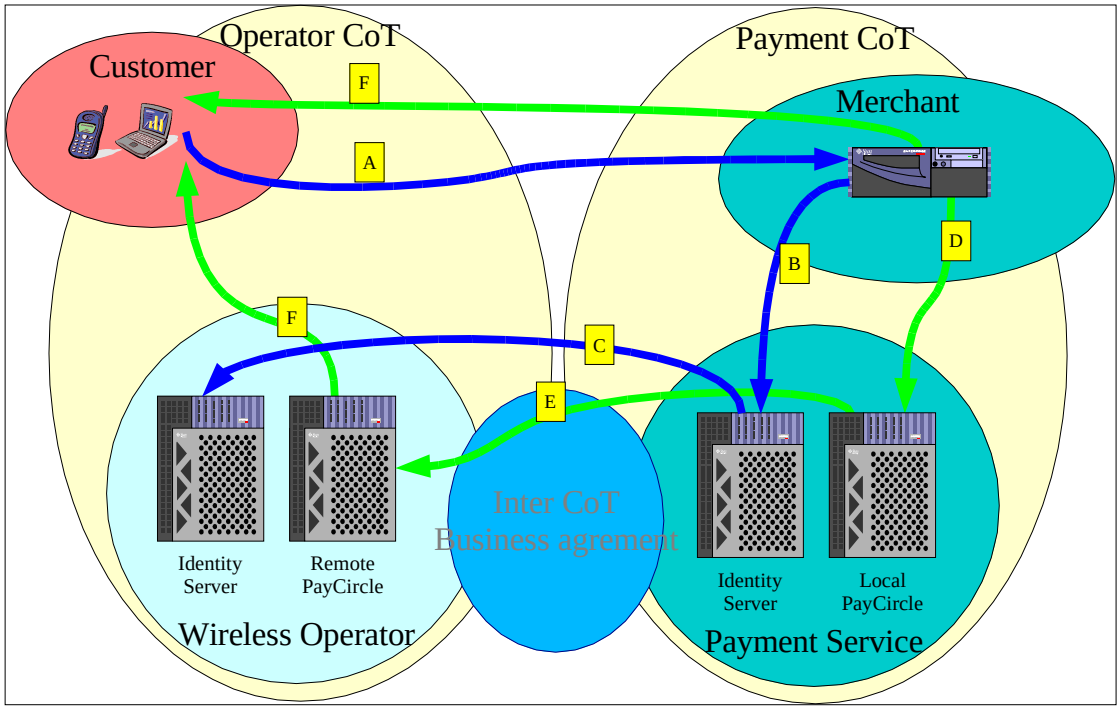
Consent request is a key element of Liberty ID-WFS, it is important to understand that while consent is requested directly by the identity service that need it, the actual dialogue with the principal may be delegated to an other entity of the architecture that get the trust of the user.

F) Deliver the service when possible [ex: game, music, ...], or return the confirmation of purchase.

3.2 4-Box Model

The main difference in between the 3-box and the 4-box model, is that in this last case the merchant is not part of customer natural circle of trust. A typical example might be user while roaming that is willing to buy something from a local shop, this model also applies to the credit card company, where merchant signs with a credit card operator and customer with a bank.

The main technical challenge of the presence of two CoT, is that when merchant try to authenticate user, while it has to delegates authentication to its own IDP, this one having no direct business relationship with the user, does not know him and thus cannot answer to the authentication request. Liberty defines a proxy authentication model that solves this challenge.



Simplified sequence flow

A) User sends request (cf. 3 box model)

B) Merchant sends an authentication request, nevertheless since it does not have a contract with the Customer IDP, it can only redirect this one onto its own IDP. At this point Payment IDP has no knowledge of the customer and cannot authenticate the user.

C) Because of business agreement in between both Payment and Operator CoT, Payment IDP can turn as a service provider of Operator CoT and thus redirect user onto operator IDP to assert the authentication (as previously IDP discovery is not described here). The result of this authentication request will be a pseudonymous/anonymous token that allows Payment IDP to create a local/roaming session inside its own context. After this is being done, nothing prevents the payment IDP to return an authentication request to merchant service provider and user is now logging onto merchant site. This mechanism is known under Liberty as proxy authentication. *Note: proxy authentication allows to propagate principal from one CoT to an other one while keeping track of Liberty constraints on privacy. In this case only the operator knows the real identity of the user, it is the only one that has a direct business agreement with this one. Payment IDP know the user as operator-customer-xxx, which is enough for requesting charging onto the operator payment interface, finally merchant only knows the user has payment-user-zzz, which is enough for requesting charging onto its own payment system. Again this step actually may include several interactions related to a payment transaction like reservation and subsequent final capture.*

D) Merchant uses bootstrap information it got from its payment service with the authentication response to discover and then invoke its payment service on behalf of the customer.

E) Remote payment service transfers payment onto customer local payment service. This exchange can be done within or outside Liberty ID-WSF, nevertheless it is

important not to forget that remote payment only knows the user through the Liberty opaque token it gets during the authentication sequence.

F) Payment service needs to ask for user consent. As previously in the 3 box model we have 3 profiles to request user consent. In redirect case, remote payment service receives a redirect request for consent from customer local payment service that it passes back to customer through merchant and user gives it consent directly to it local payment service. We may imagine that in case of foreign wireless operator or bank acting as remote payment, an option of having the remote payment to handle consent request directly might be accepted by customer because they are viewed as trusted entities. Obviously other models remains valid and especially the independent interaction server registration within customer discovery service. It is up for further study to check whether the redirect model would work in this scenario.

G) Return of final payment status.

3.3 Additions Needed in the PayCircle WSI

Details are for further study.

1. It has to be specified, that HTTP headers have to be Liberty compliant
2. WSI has to include resource ID
3. WSI has to include Error for “consent needed” (in redirect models)

3.4 Additions Needed in Payment System Implementations

Details are for further study.

1. Invocation of interaction service (only needed for intercation service model, obsolete for redirect model)
2. Mapping of resource id onto subscriber id (at PSP in 3-box)
3. Mapping of resource id onto different resource id (at acquirer) and onto subscriber id (at issuer) (in 4-box)

3.5 Additions Needed in Liberty Specifications

Details are for further study.

3.6 Additons Needed in Identity Service Implementations

Details are for further study.

4 Open Issues

The following issues have not been addressed in this document version

1. Mechanisms for signing user consent, this includes terminal capabilities
2. Security, WS encryption

5 Glossary

Acquirer

The payment service provider of the merchant.

Attribute Provider (AP)

The attribute provider (AP) provides Identity Personal Profile (ID-PP) information. Sometimes called an ID-PP provider, the AP is a ID-WSF web services that hosts the ID-PP. See SiS

Circle of Trust (CoT)

A federation of service providers and identity providers that have business relationships based on Liberty architecture and operational agreements and with whom users can transact business in a secure and apparently seamless environment. Circles of Trust represent the second wave of identity federation, after SSO and federated account linking.

Discovery Service (DS)

A Liberty service for locating attribute providers. It defines a core identity service that enables various entities, such as SPs, to dynamically discover a user's registered identity services. Given the type of service desired (e.g., Person Identity Profile Service), the Discovery Service responds, per the appropriate permissions, with a service description containing a Web Services Definition Language (WSDL) profile for the desired identity service.

End point

A SOAP (RPC) address and function name that can be used to obtain some service. In Liberty entry points are what a Discovery Service allows one to discover.

Federate

To link or bind two or more entities together. See Identity Federation.

Identity

A single individual, or principal, often has multiple identities. An identity is a set of attributes associated with a user's various accounts on various domains, sites, or applications throughout the Internet. An identity encompasses such attributes as the individual's proper name, e-mail address, credit card number, Social Security number, driver's license, buying history, and notification preferences.

Identity federation

An individual may choose to link his or her various identities across diverse IdPs and SPs through a process called federation. Identity federation is the linking of two or more accounts that are associated with a given principal across various Liberty enabled entities within a Circle of Trust.

Identity Provider (IdP)

A Liberty-enabled entity that creates, maintains, and manages identity information for Principals and provides Principal authentication to other service providers within a circle of trust. Typically, a Liberty-enabled IdP is a Web portal through which a user logs into a federated application environment. An IdP will usually have an associated portal interface, authentication service, and user directory. The IdP incorporates a SAML Authentication Authority, and after authenticating

user credentials against a trusted authentication service, transmits SAML Authentication Assertions to the SPs that control access to the resources that users are requesting.

Issuer

The payment service provider of the subscriber/customer.

Liberty-enabled Provider

Liberty-enabled Provider may be either an Attribute Provider (AP), Discovery Service (DS), Service provider (SP), Identity Provider (IdP) who collects, transfers, or receives the Personally Identifiable Information (PII) of a Principal.

Payment Service Provider (PSP)

The payment service provider is an actor who enables merchants and customers to conduct payments with each other.

Principal

A Principal is an entity that can acquire a federated identity, that is capable of making decisions, and to which authenticated actions are done on its behalf. Examples of principals include an individual user, a group of individuals, a corporation, other legal entities, or a component of the Liberty architecture.

Pseudonym (pseudonymity)

An arbitrary name assigned by the identity or service provider to identify a Principal to a given relying party so that the name has meaning only in the context of the relationship between the relying parties.

SAML (Security Assertion Markup Language)

The Security Assertion Markup Language (SAML) standard defines a framework for exchanging security information between online business partners. It was developed by the Security Services Technical Committee (SSTC) of the standards organization OASIS (the Organization for the Advancement of Structured Information Standards). More precisely, SAML defines a common XML framework for exchanging security assertions between entities.

Service Provider (SP)

An individual's identities are used by networked functional entities called SPs when authenticating and authorizing the user to access resources controlled by SPs. An SP is a Liberty-enabled infrastructure entity that controls access to resources requested by users and relies on SAML Authentication Assertions issued by IdPs. An SP may maintain user information for profiling purposes but leave the credentials management and authentication functions to the user s IdP.

Typically, a Liberty-enabled SP is another portal or Web-based application (managed separately from the IdP) that controls access to resources that require user authentication. An SP will usually have an associated portal interface, authorization service, and policy rule base. The SP incorporates a SAML Policy Decision Point (PDP) and Policy Enforcement Point (PEP), relying on Authentication Assertions created and transmitted from IdPs.

Single sign-on (SSO)

The ability to use proof of an existing authentication session with identity provider A to create a new authentication session with identity provider B.

SOAP (Simple Object Access Protocol)

An XML envelope and data encoding technology used to communicate information and requests across the Web. It is typically considered the protocol used by Web services. It is actually an envelope encapsulation format that can be used with lower level Web protocols such as HTTP and FTP.

SiS

Service Interface Specification of the Liberty Alliance Project.

User Agent

Any software that retrieves and renders Web content for users.

Web service

A service that uses Internet protocols to provide a service designed to be used by programs.

Web Service Consumer (WSC)

An entity that uses a web service to access data.

Web Service Provider (WSP)

An entity that provides data via a web service.