SAML V2.0 Basics

Eve Maler
eve.maler@sun.com
Sun Microsystems, Inc.

Updated 2 October 2006

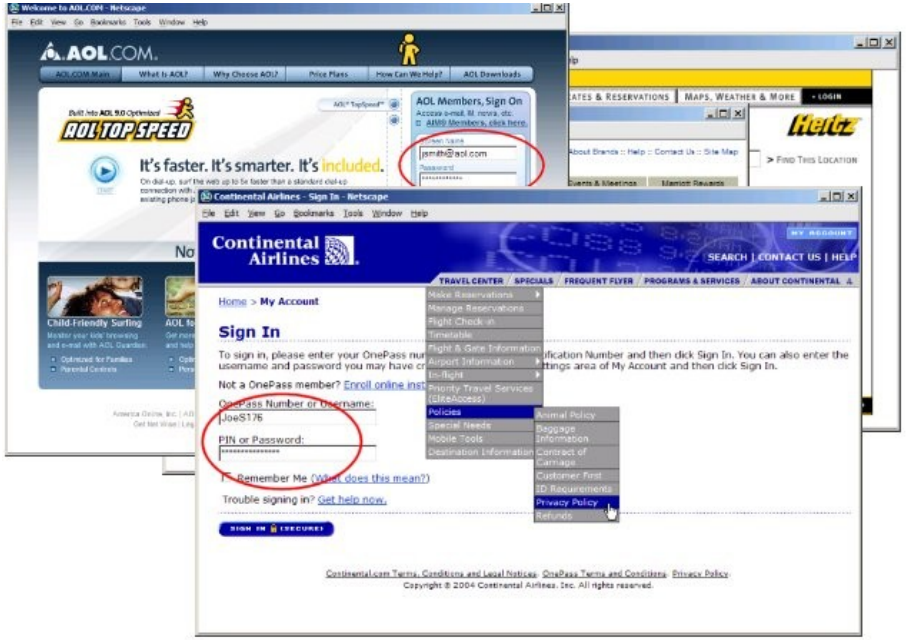
Topics

- The big picture
- The standards landscape
- SAML concepts and terms
- SAML assertions
- Major SAML usage scenarios
- How you can get started
- Resources

The big picture

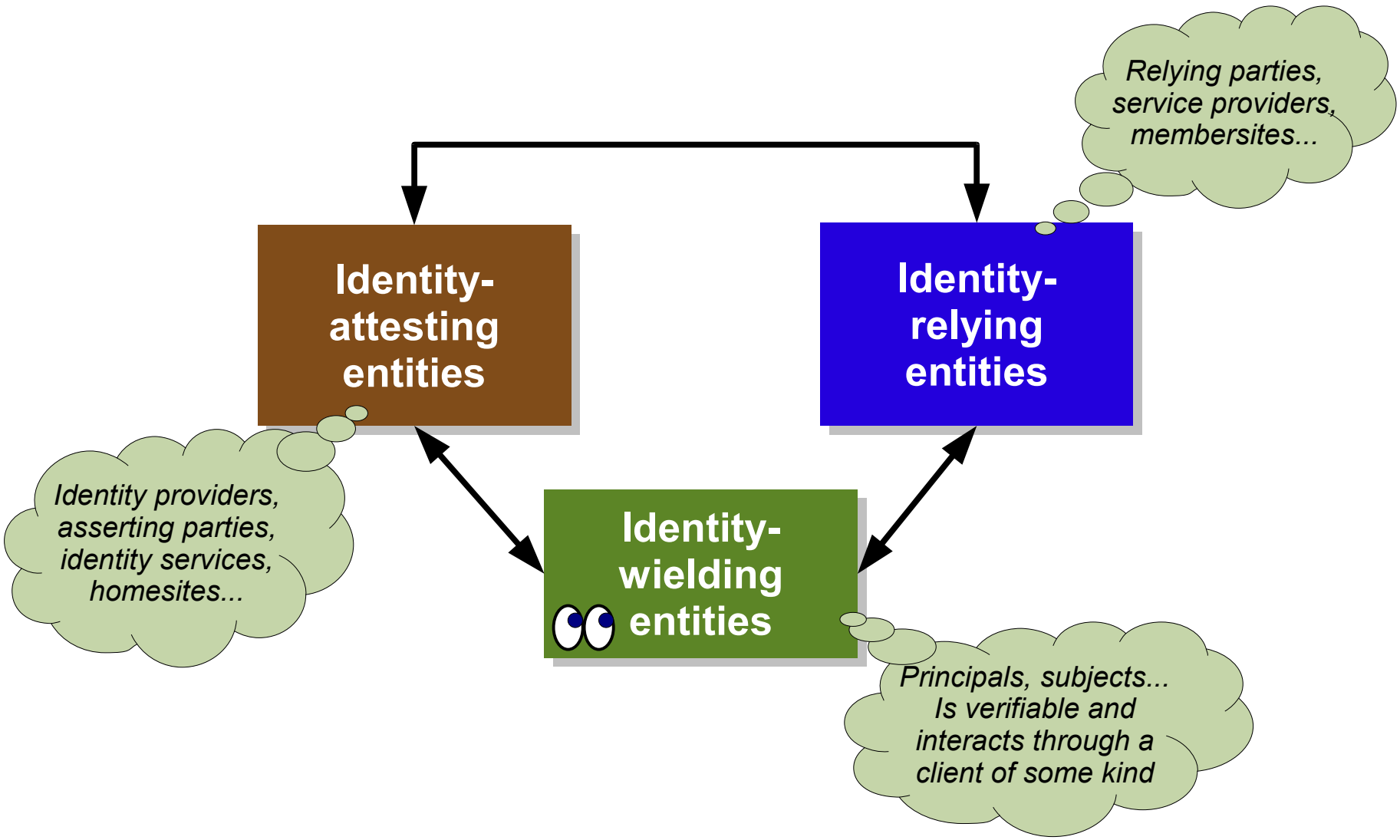
Opportunities with distributed identity

- People can:
  - Avoid authenticating repeatedly
  - Unify management of their identity information
  - Have better-personalized online experiences
  - Gain better privacy control

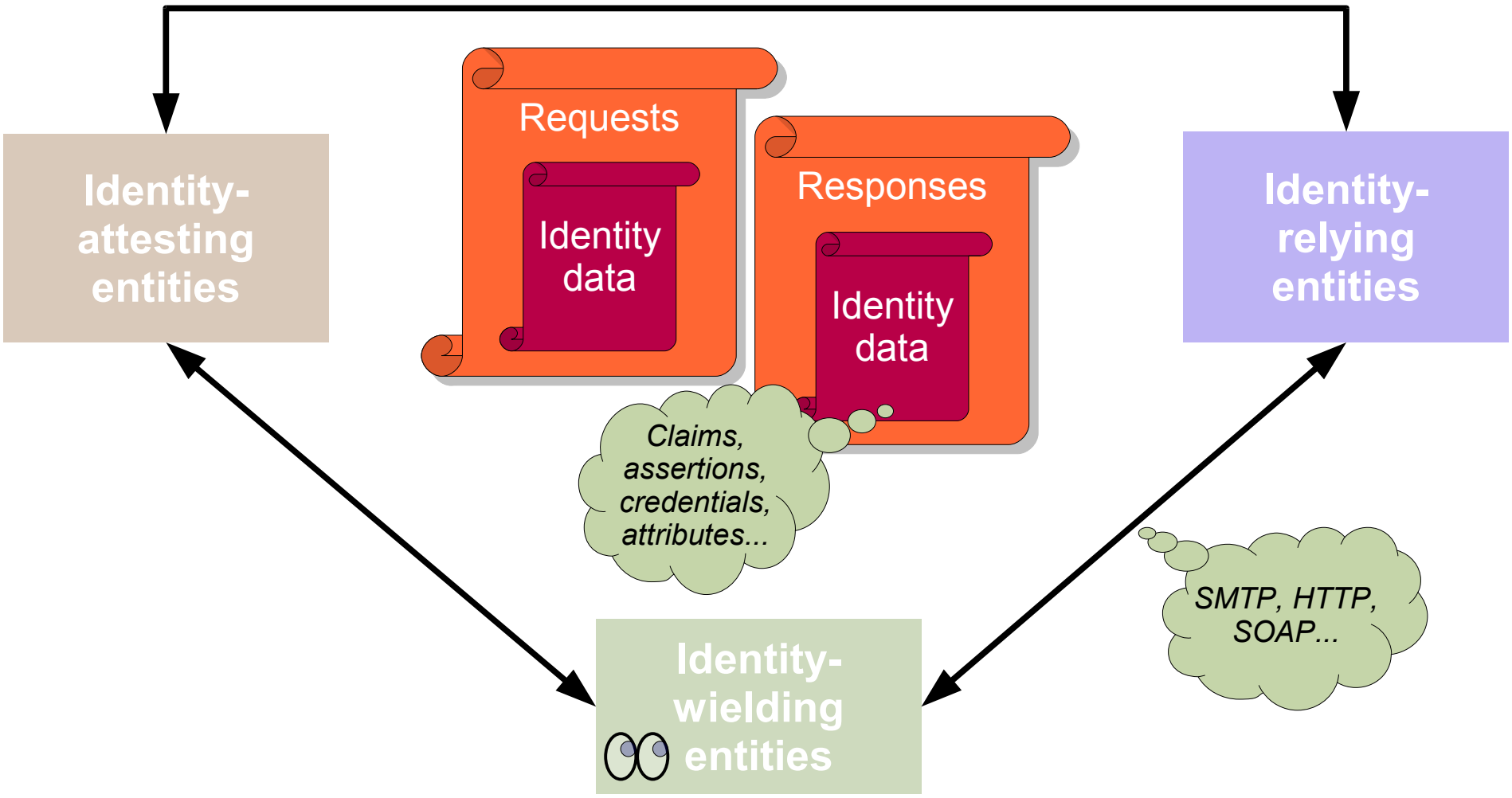


- Services and applications can:
  - Offload authentication and identity lookup tasks
  - Unify treatment of all "things with identities"
  - Provide finer-grained access control and differentiation
- Organizations can:
  - More securely outsource business functions

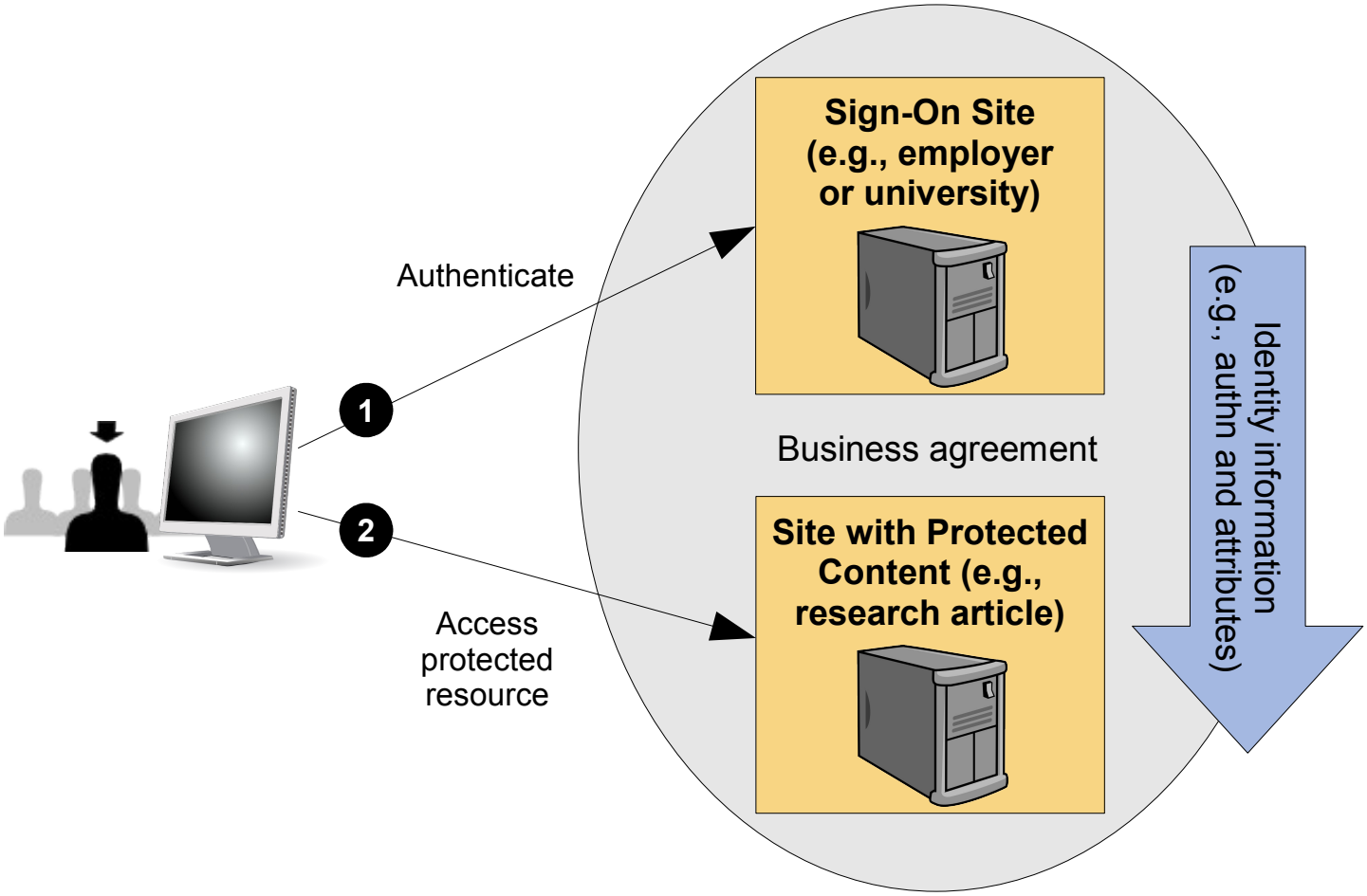
Essential roles in a distributed identity architecture



Identity data distribution flows and containers



The basic use case for single sign-on (SSO)

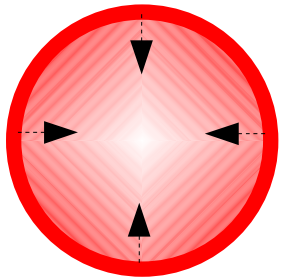


Fleshing out a scenario: web-based IM

- Specialized web IM application in a manufacturing environment
  - Used to notify repair personnel about, and let them discuss, equipment breakdown episodes
- Employees have IM access by virtue of:
  - A valid **login** to the company portal
  - A **role** of "repair_tech"
- An identity service can ping all online repair techs automatically to discuss malfunction triage situations
- Employees can log out of their portal, IM, and all their other work apps in one step

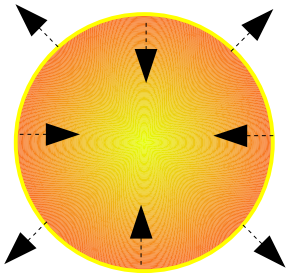
Technical challenges with distributed identity

- Distributing identity info across domain boundaries in the first place – privacy, security, accuracy, compliance...



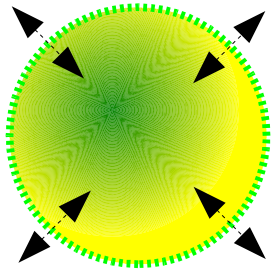
The Era of the Firewall

Keep data inside the firewall



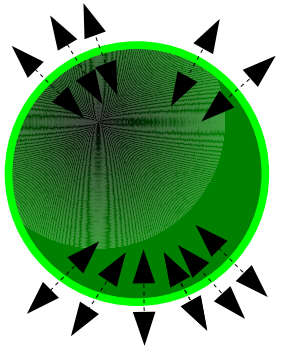
The Era of the Intranet/Internet

Manage data inside and outside the firewall



The Era of the Extranet

Manage data through the firewall



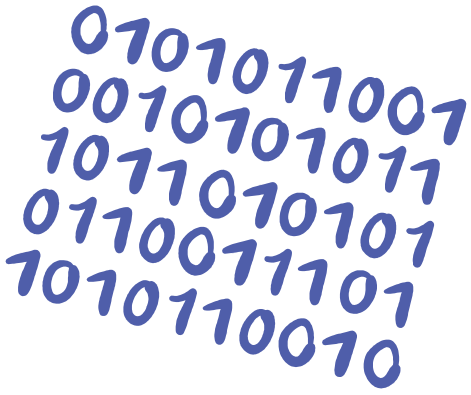
Nothing But Net

Just access and entitlement for identity wielders

- Getting the identity info semantics right – the syntax is the comparatively easy part
- Security solutions at the application layer never absolve you from providing security below

Requirements for distributed identity

- Standard, flexible **formats** for identity information
- **Protocols** that are standard, secure, privacy-enabled, technology-neutral, and interoperable for exchanging identity information between components of distributed applications
- A way to set up **trust relationships** between entities that share identity information within technical, business, and legal frameworks



The standards landscape

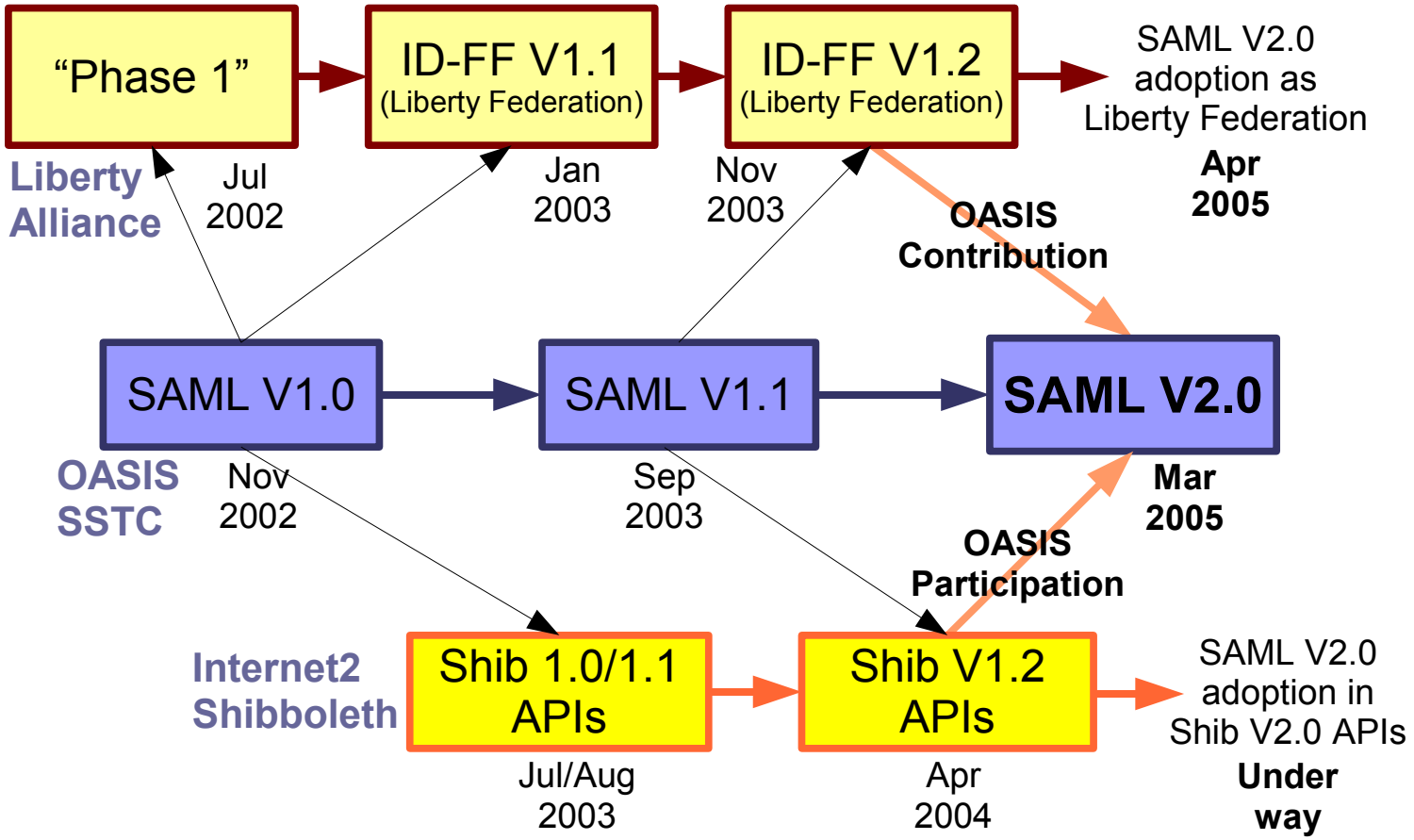
The overall identity landscape (YMMV)

- Milestones compiled by Internet Identity Workshop 2 participants in May 2006: http://photos.windley.com/albums/iiw2006a/IIW2006_identity_map

Focusing on SAML

- The Security Assertion Markup Language in six words:
  "The universal solvent of identity information"
- Best supported and most thoroughly standardized, covering a wide range of distributed-identity scenarios
  - Reflects the convergence of several development streams
  - Enables privacy along various dimensions
- Many other specs and standards build on it

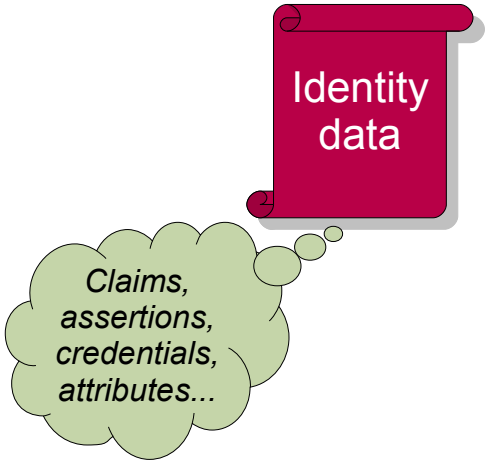
Liberty / SAML / Shibboleth: one degree of separation



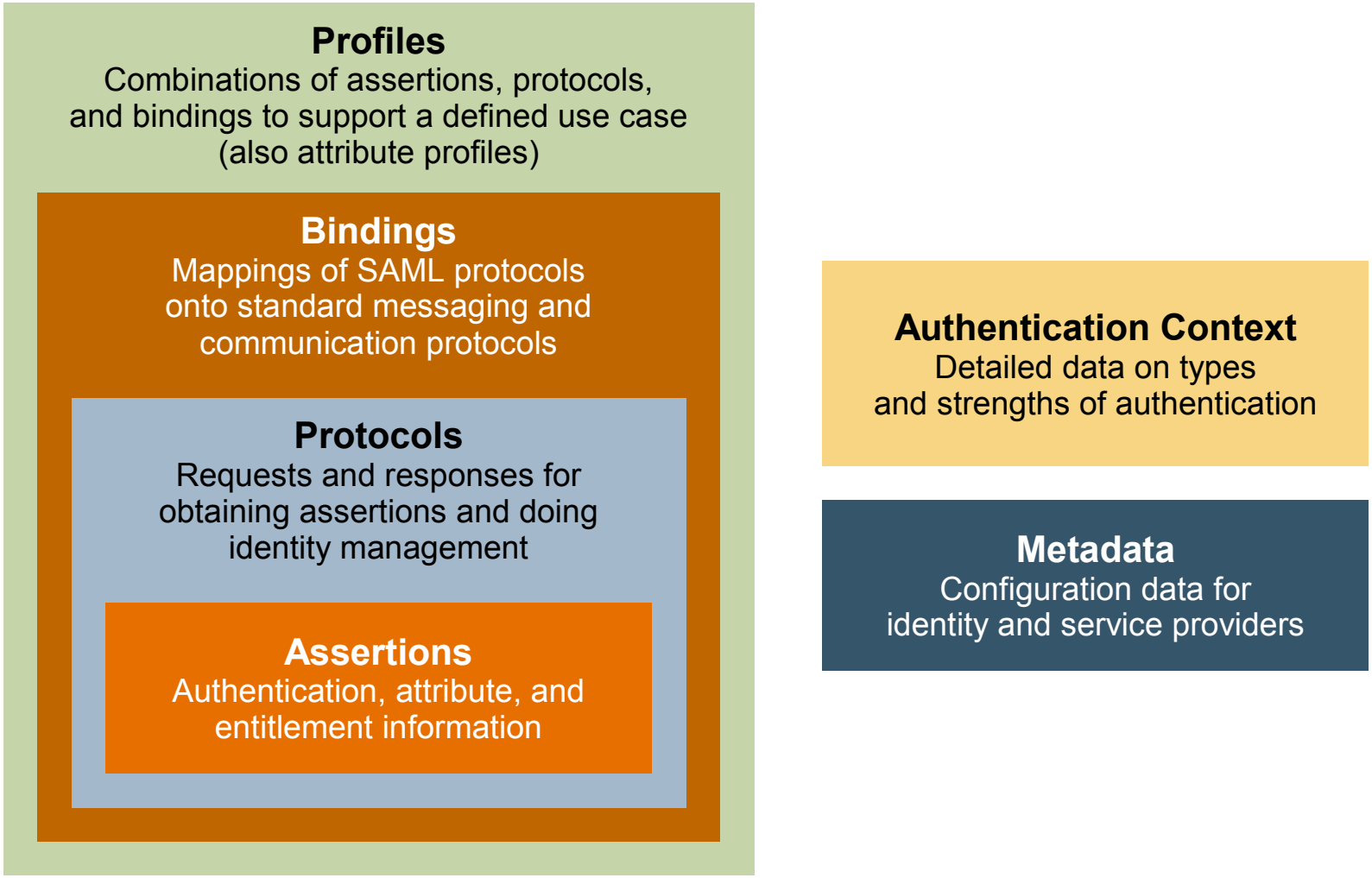
SAML concepts and terms

SAML in a technical nutshell

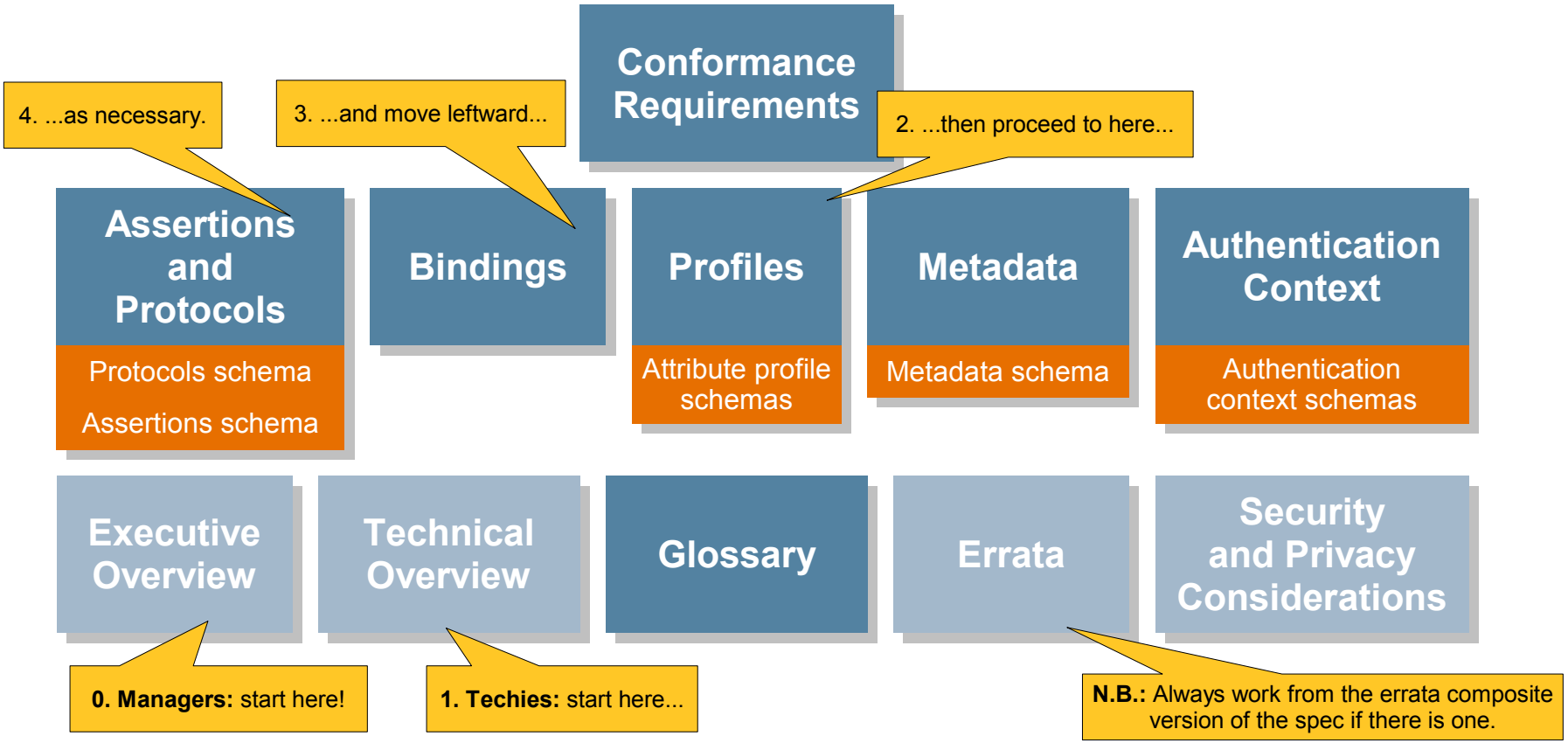
- SAML in 15 words: ***"XML-based framework for marshaling security and identity information and exchanging it across domain boundaries"***
  - It wraps existing security technologies rather than inventing new ones
  - Its profiles offer interoperability for a variety of use cases, but you can extend and profile it further
- At SAML's core:
  assertions about subjects
  - Authentication, attribute, entitlement, or roll-your-own



SAML components and how they relate to each other

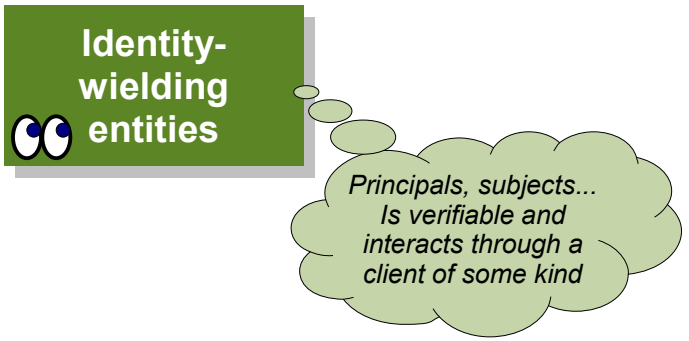


The SAML specifications map to them fairly closely



Language about subjects

- **Entity (or system entity):** An active element of a computer/network system
- **Principal:** An entity whose identity can be authenticated
- **Subject:** A principal in the context of a security domain



Language about identities

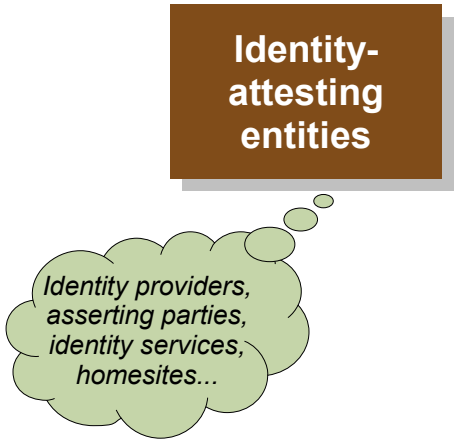
- **Identity:** The essence of an entity, often described by one's characteristics, traits, and preferences
  - **Anonymity:** Having an unknown/concealed identity
- **Identifier:** A data object that uniquely refers to a particular entity
  - **Pseudonym:** A privacy-preserving identifier

(More) language about identities

- **Federated identity:** Existence of an agreement between providers on a set of identifiers and/or attributes to use to refer to a principal
  - **Account linkage:** Relating a principal's accounts at two providers so they can communicate about it

Language about (more) entities

- **Asserting party (SAML authority):** An entity that produces SAML assertions
  - **Identity provider:** An entity that creates, maintains, and manages identity information for principals and provides principal authentication to other service providers



(More) language about (more) entities

- **Relying party:** An entity that decides to take an action based on information from another system entity
  - **Service provider:** An entity that provides services to principals or other entities

SAML assertions

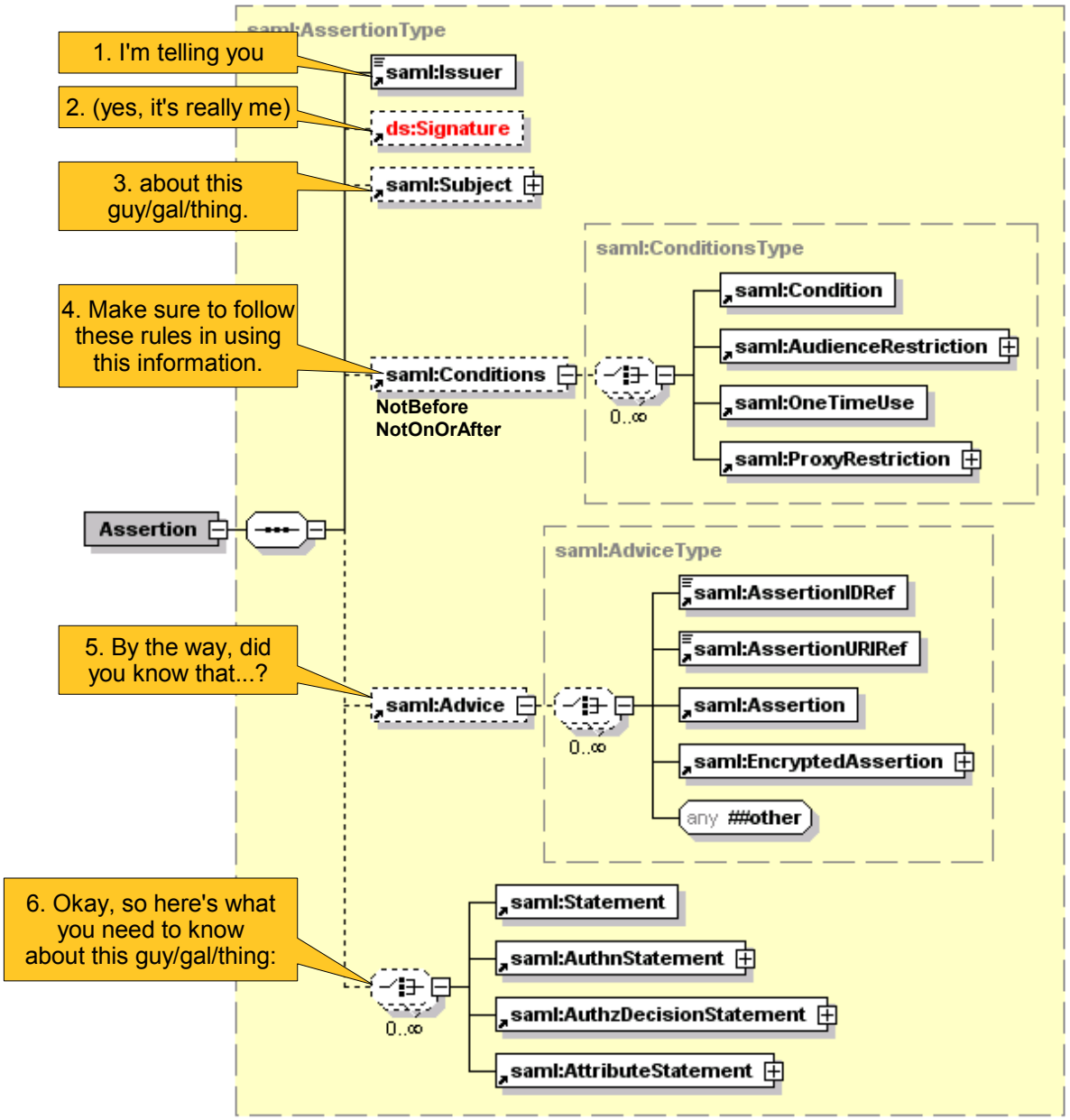
Assertion basics

- An assertion is a claim made by someone about someone
- SAML assertions are structured as a series of statements about a subject:
  - **Authentication** statement: "Sam authenticated with a smartcard PKI certificate at 9:07am today"
  - **Attribute** statement (which can contain multiple attributes): "Sam is a manager and has a $5000 spending limit"
  - **Authorization decision** statement (now deprecated): "Yes, Sam can read that web page"
  - Your own customized statements...

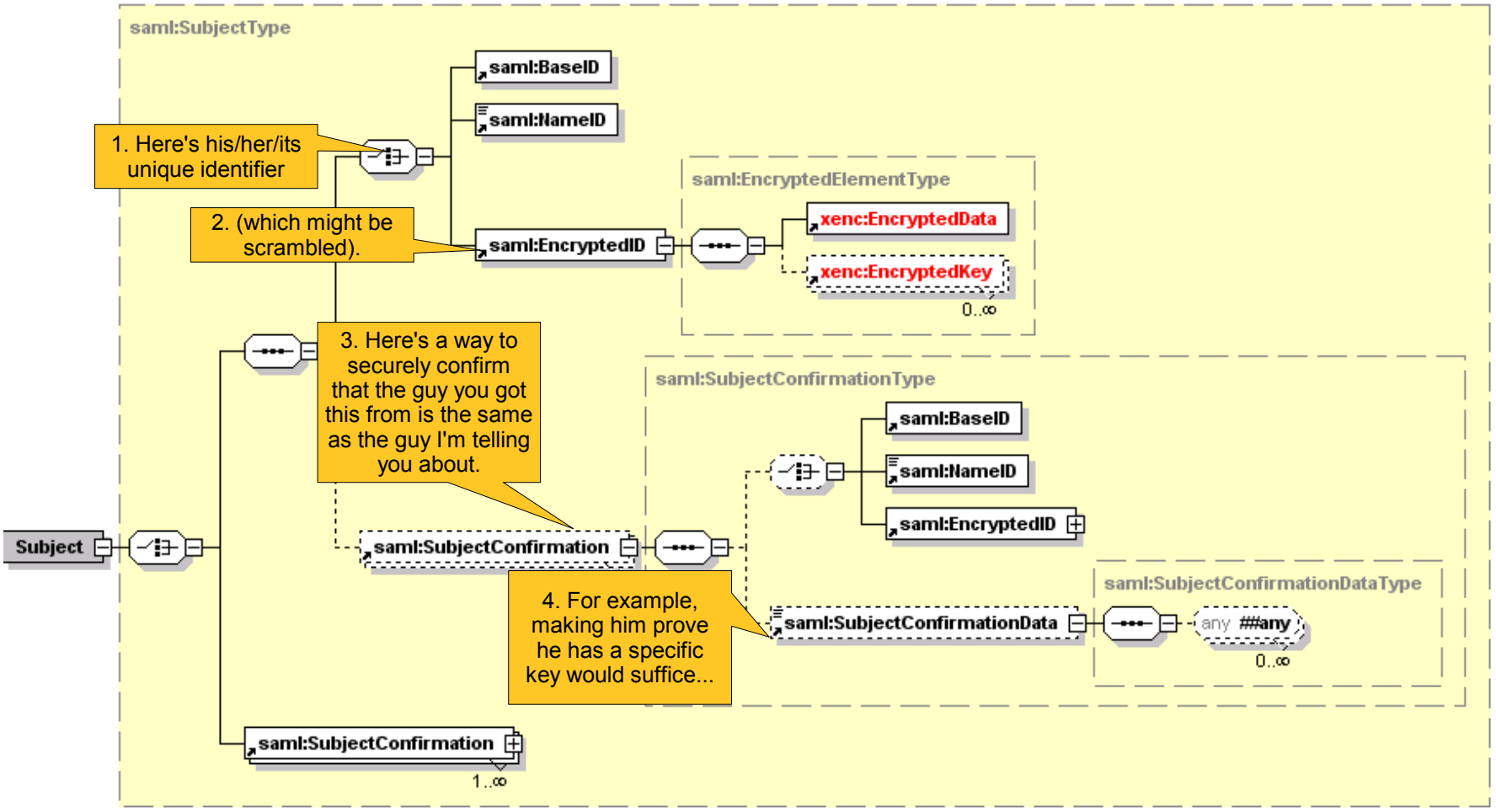
Example of an assertion's common portions

```
<saml:Assertion
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  Version="2.0"
  IssueInstant="2006-07-28T14:01:00Z">
  <saml:Issuer>
    www.emeffgee.com
  </saml:Issuer>
  <saml:Subject>
    <saml:NameID
  Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
      J.Handy@emeffgee.com
    </saml:NameID>
  </saml:Subject>
  <saml:Conditions
    NotBefore="2006-07-28T14:00:05Z"
    NotOnOrAfter="2006-07-28T14:05:05Z">
  </saml:Conditions>
    ... statements go here ...
</saml:Assertion>
```

Overall assertion element structure



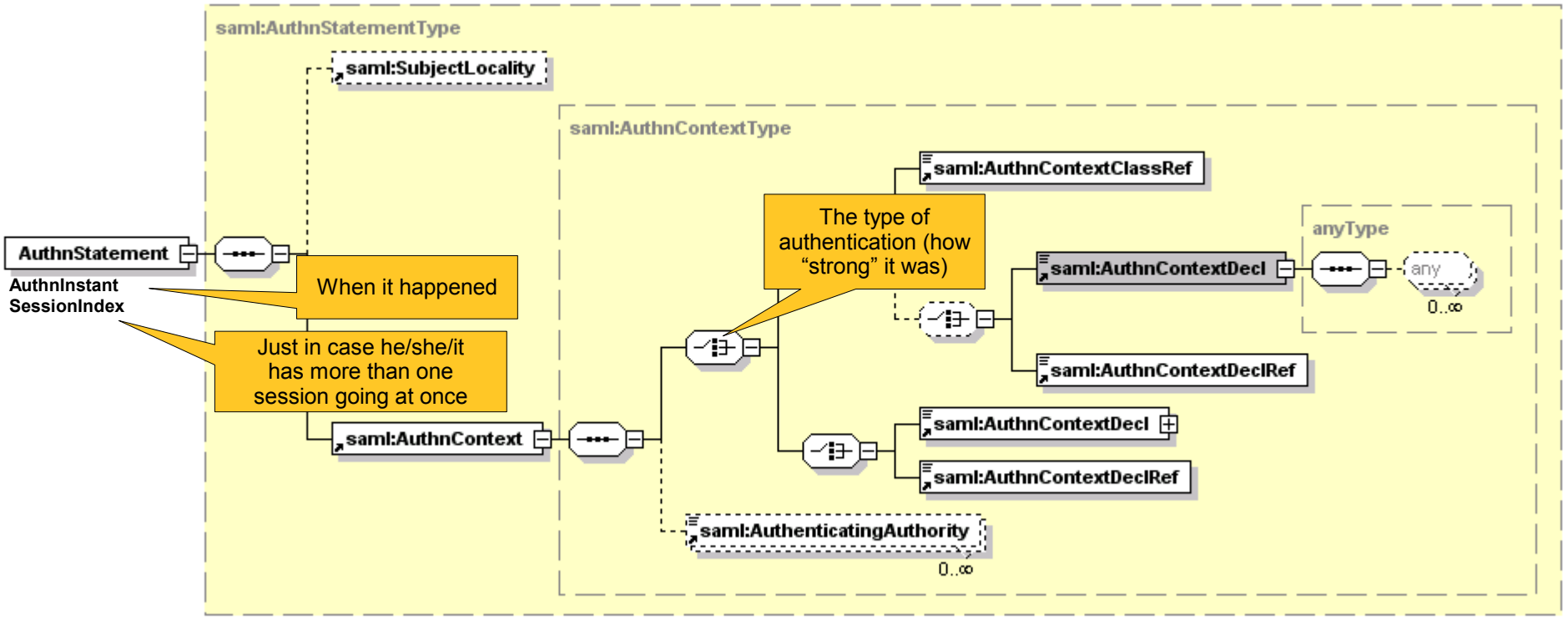
Subject element structure



Example of an authentication statement

```
<saml:Assertion ... common info goes here ... >
  ... and here ...
  <saml:AuthnStatement
    AuthnInstant="2006-07-28T14:00:05Z"
    SessionIndex="0">
    <saml:AuthnContext>
      <saml:AuthnContextClassRef>
urn:oasis:names:tc:SAML:2.0:ac:classes:SmartcardPKI
      </saml:AuthnContextClassRef>
    </saml:AuthnContext>
  </saml:AuthnStatement>
  ... more statements might appear here ...
</saml:Assertion>
```

Authentication statement element structure



Authentication context classes

SAML comes with a healthy set of predefined identifiers for typical authentication scenarios:

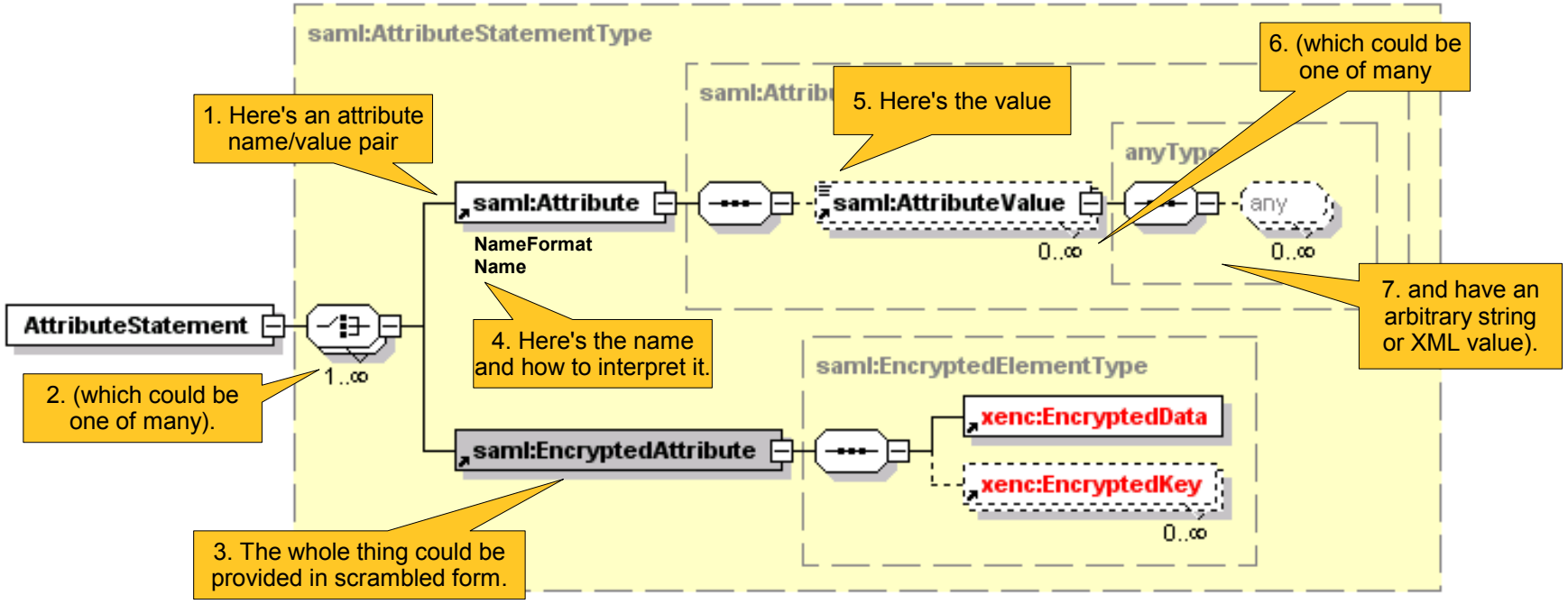
- Internet Protocol
- Internet Protocol Password
- Kerberos
- Mobile One Factor Unregistered
- Mobile Two Fa1ctor Unregistered
- Mobile One Factor Contract
- Mobile Two Factor Contract
- Password
- Password Protected Transport
- Previous Session
- Public Key – X.509
- Public Key – PGP
- Public Key – SPKI
- Public Key – XML Signature
- Smartcard
- Smartcard PKI
- Software PKI
- Telephony
- Nomadic Telephony
- Personalized Telephony
- Authenticated Telephony
- Secure Remote Password
- SSL/TLS Cert-Based Client Authentication
- Time Sync Token
- Unspecified

You can also create or customize your own authentication context classes...

Example of an attribute statement

```
<saml:Assertion ... common info goes here ... >
  ... and here ...
  <saml:AttributeStatement>
    <saml:Attribute
      NameFormat="http://emeffgee.com" Name="Role" >
      <saml:AttributeValue>repair_tech</saml:AttributeValue>
    </saml:Attribute>
    <saml:Attribute
      NameFormat="http://emeffgee.com">
      Name="Certification"
      <saml:AttributeValue xsi:type="emeffgee:type">
        <emeffgee:CertRecord language="EN">
          <Course>
            <Name>Structural Repair</Name>
              <Credits>3</Credits>
          </Course> ...
        </emeffgee:CertRecord>
      </saml:AttributeValue>
    </saml:Attribute>
  </saml:AttributeStatement>
</saml:Assertion>
```

Attribute statement element structure



Attribute profiles

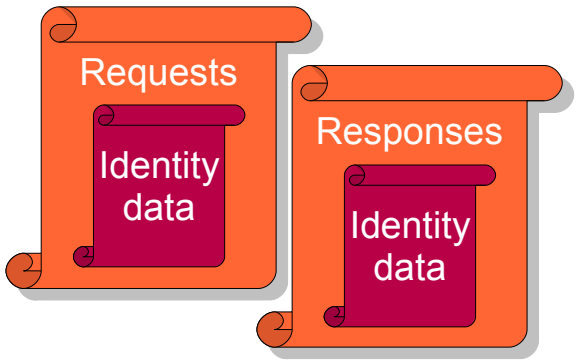
- Basic
  - Simple string-based SAML attribute names
- X.500/LDAP
  - Common standardized convention for SAML attribute naming using OIDs, expressed as URNs
- UUID
  - SAML attribute names as UUIDs, expressed as URNs
- DCE PAC
  - Representation of DCE realm, principal, and group membership information in SAML attributes
- XACML
  - How to map SAML attributes cleanly to XACML attribute representation
- XPath (draft)
  - XPath expression pointing to the attribute values within an XML document as the attribute name – has utility in identity services
- Your own customized attribute profiles...

So far, no interchange – just format

- SAML assertions are becoming *the* way to marshal packets of identity information
  - They wrap existing authentication and attribute (and authorization) semantics rather than inventing new ones
- Getting them from point A to point B has two interesting aspects:
  - **Why?** What purpose is being served in sending and getting them?
  - **How?** Along what channels do they flow?
  - There are security and privacy implications for both

Request/response protocols

- Assertions are requested, provided as input, and returned as output in the course of doing these jobs



- SAML defines various XML request/response **protocol** message pairs
  - All based on a hierarchy of complex data types in the protocol schema
- The messages can be conveyed using various communications protocols through SAML **bindings**

Major SAML usage scenarios

Key use cases covered by SAML profiles

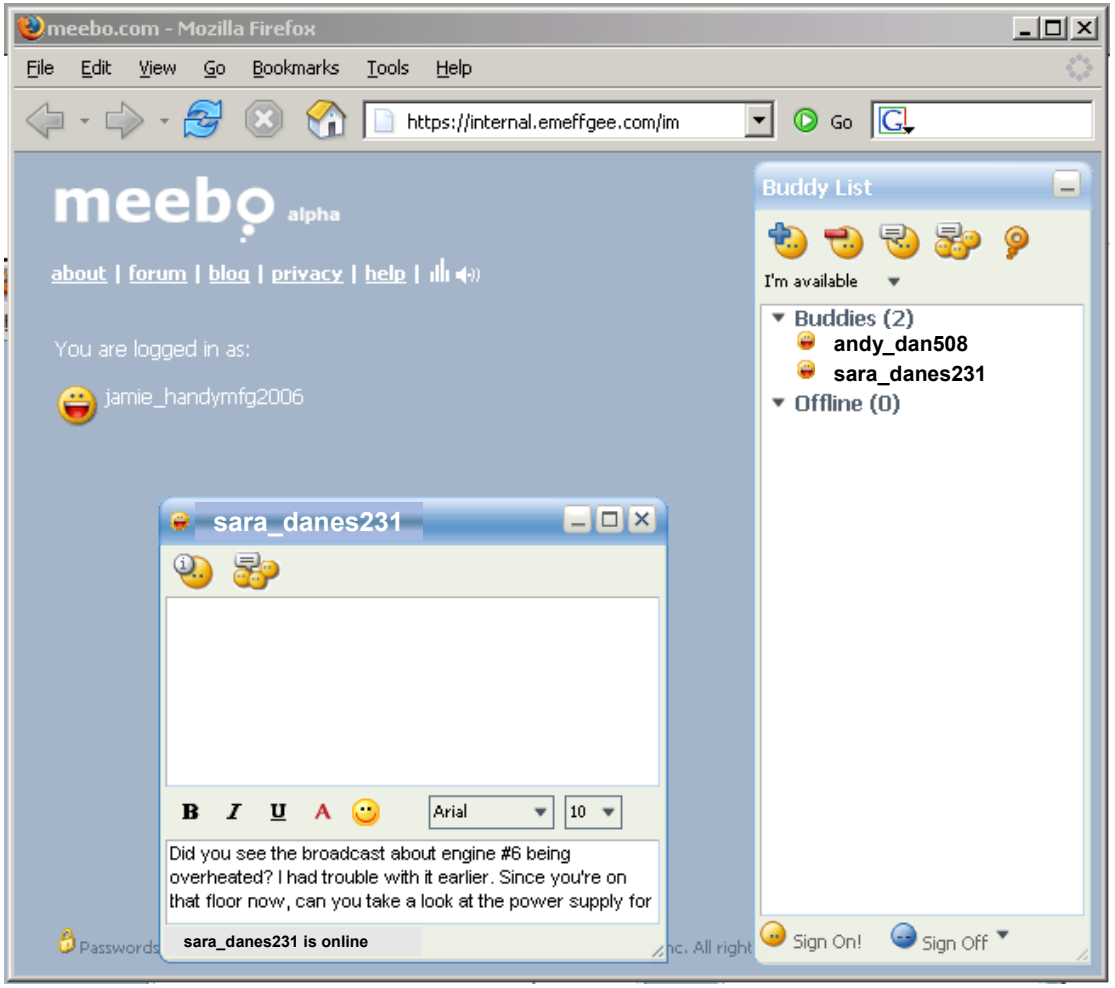
- Single sign-on
  - Using standard browsers and enhanced clients (such as handheld devices)
- Federating identities
  - Using a well-known identifier or a privacy-preserving pseudonym
- Attribute services
- Single logout
- You can create your own profiles...
  - E.g., WS-Security defines a SAML Token Profile for securing web services

The vanilla Web SSO profile

- Goal: J. Handy, repair tech, signs in only once whenever using the company portal *and* the IM app
- Requirement: The portal has to prove to the IM app that J. is authenticated, and also provide attributes that will let the portal make an authorization decision
- The players:

A (mockup of a)n IM conversation

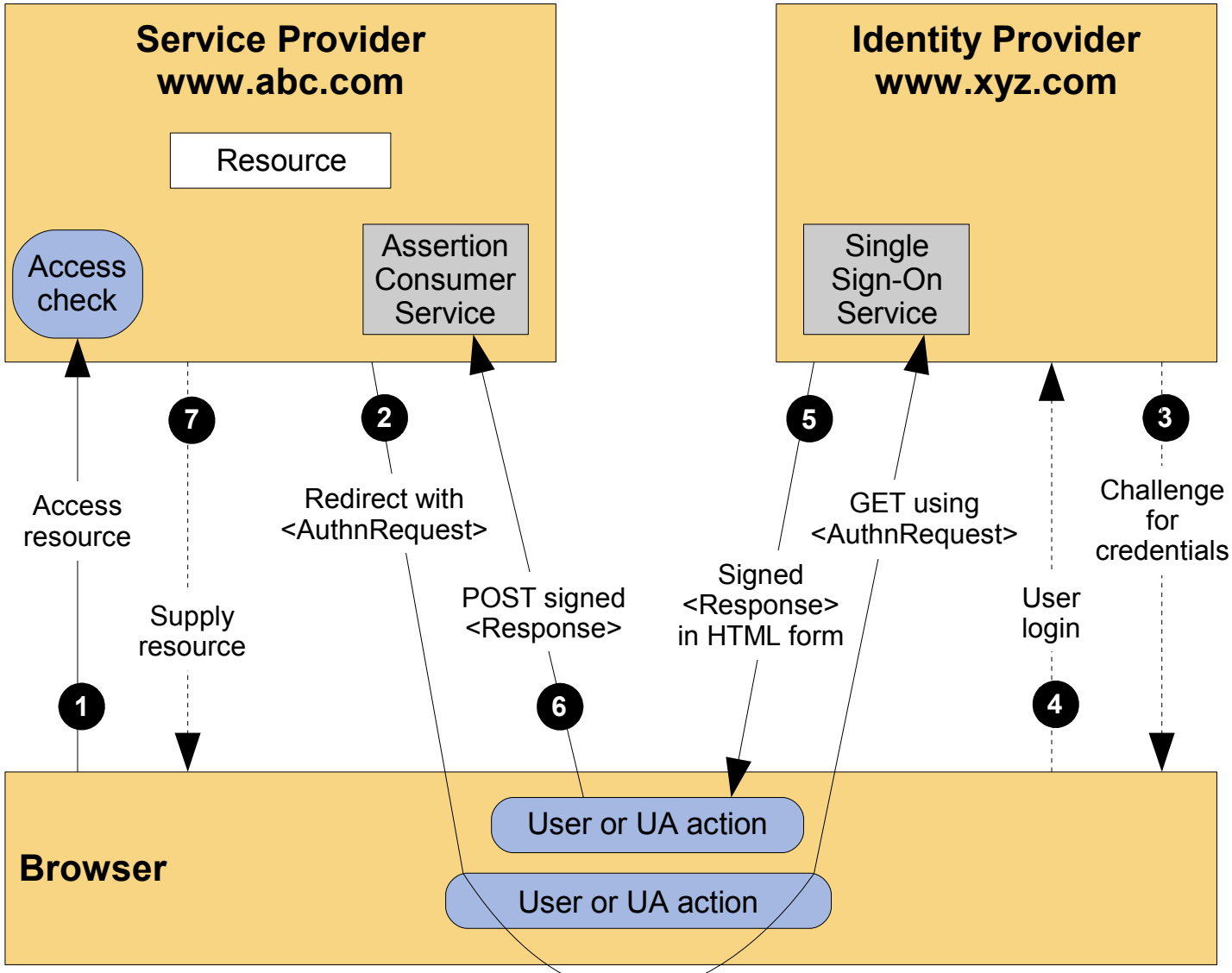
- J. Handy has logged in either at the IM app prompt or the company portal
- Engine #6 itself alerted all online techs to the overheating
- Jamie has determined through presence/location services where the other techs are
- Meebo might be providing infrastructure that EmEffGee hosts itself, or alternatively, it's an outsourced service



Several options for information flow

- Does J. visit the portal or the IM app first?
  - If J. tries to use IM first, the IM app has to explicitly **request** info from the portal
  - The SSO assertion has to be conveyed from IdP to SP regardless, using a **response** message
- If the IM app makes a request, does it push (HTTP POST), allow to be pulled ("artifact"), or use HTTP redirect for the request?
- Does the portal push (HTTP POST) or allow to be pulled ("artifact") the response?
- Let's see...carry the two...that's eight options
  - But some are more common than others

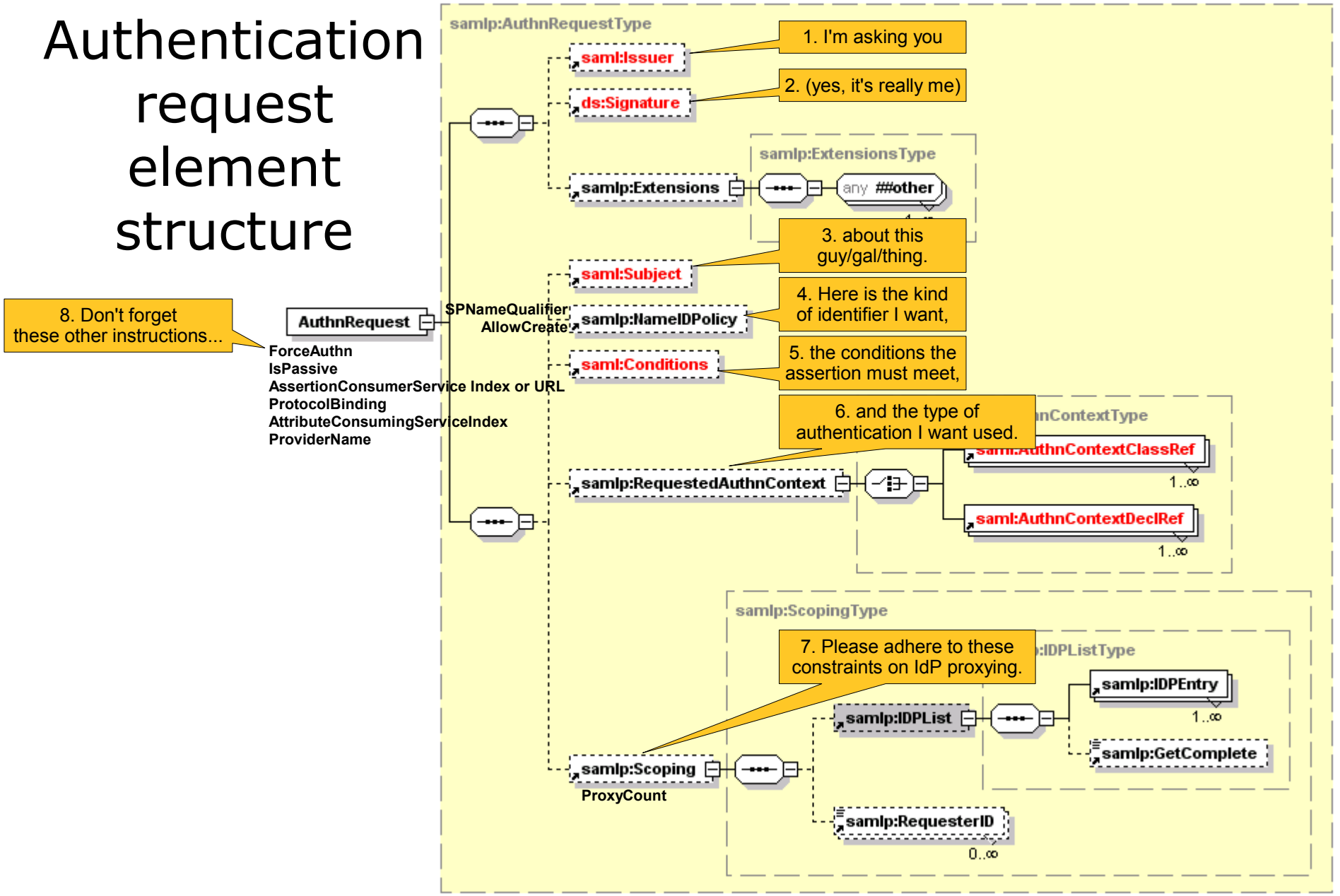
SP-initiated flow with redirect and POST bindings



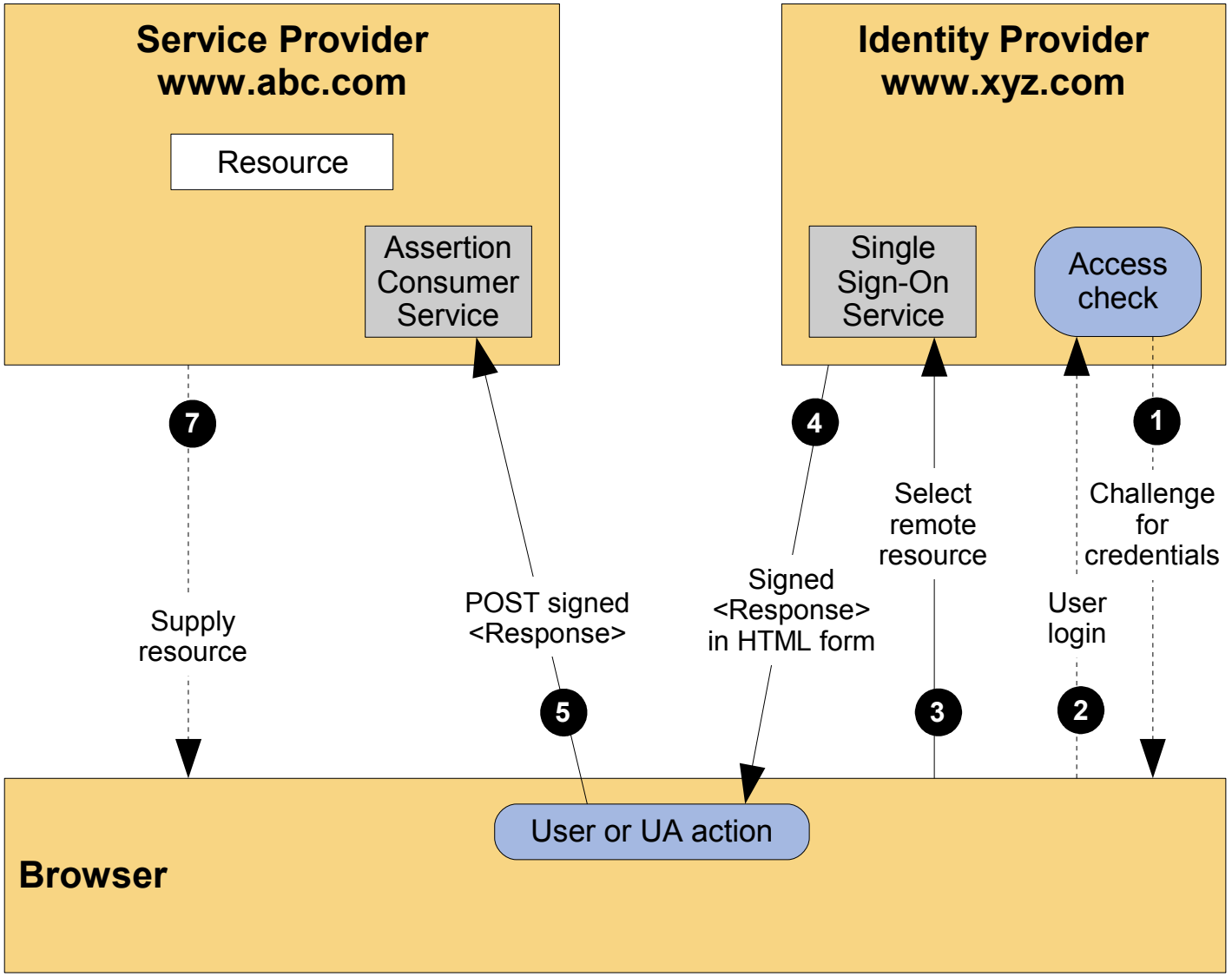
Example of an authentication request

```
<samlp:AuthnRequest
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" Version="2.0"
  ID="f0485a7ce95939c093e3de7b2e2984c0"
  IssueInstant="2006-07-28T14:01:05Z"
  Destination="https://www.emeffgee.com/IdP/"
  AssertionConsumerServiceIndex="1"
  AttributeConsumingServiceIndex="0">
  <saml:Issuer>http://www.emeffgee.com/IM/</saml:Issuer>
  <samlp:RequestedAuthnContext>
    <saml:AuthnContextClassRef>
      urn:oasis:names:tc:SAML:2.0:ac:classes:Password
    </saml:AuthnContextClassRef>
  </saml:RequestedAuthnContext>
  <samlp:NameIDPolicy
    Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
  </samlp:NameIDPolicy>
</samlp:AuthnRequest>
```

Authentication request element structure



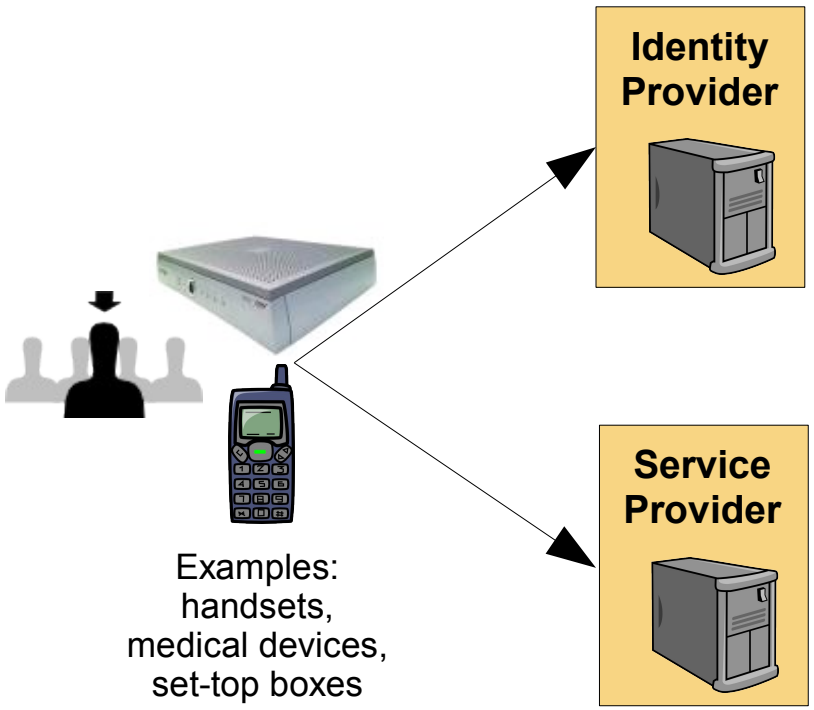
IdP-initiated flow with the POST binding



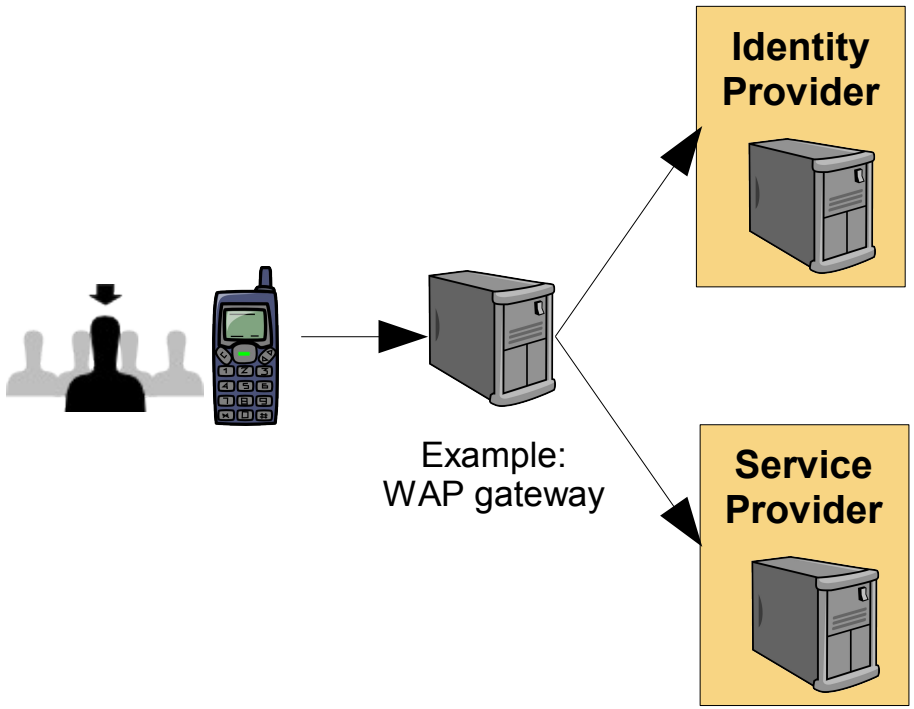
SSO using an enhanced client

- SAML defines an "Enhanced Client or Proxy" SSO profile for:
  - Proxy servers such as WAP gateways in front of limited-ability mobile devices
  - Clients that can't use HTTP redirects
  - Accommodating the inability of the IdP and SP to communicate, for whatever reasons
    - It might be an architectural choice
- In some circumstances the web client is smarter than the average bear
  - An ECP client can use the PAOS binding to communicate cleverly via SOAP and HTTP
  - It may also be clever about where to find IdPs
  - It can even be an IdP

ECP use cases

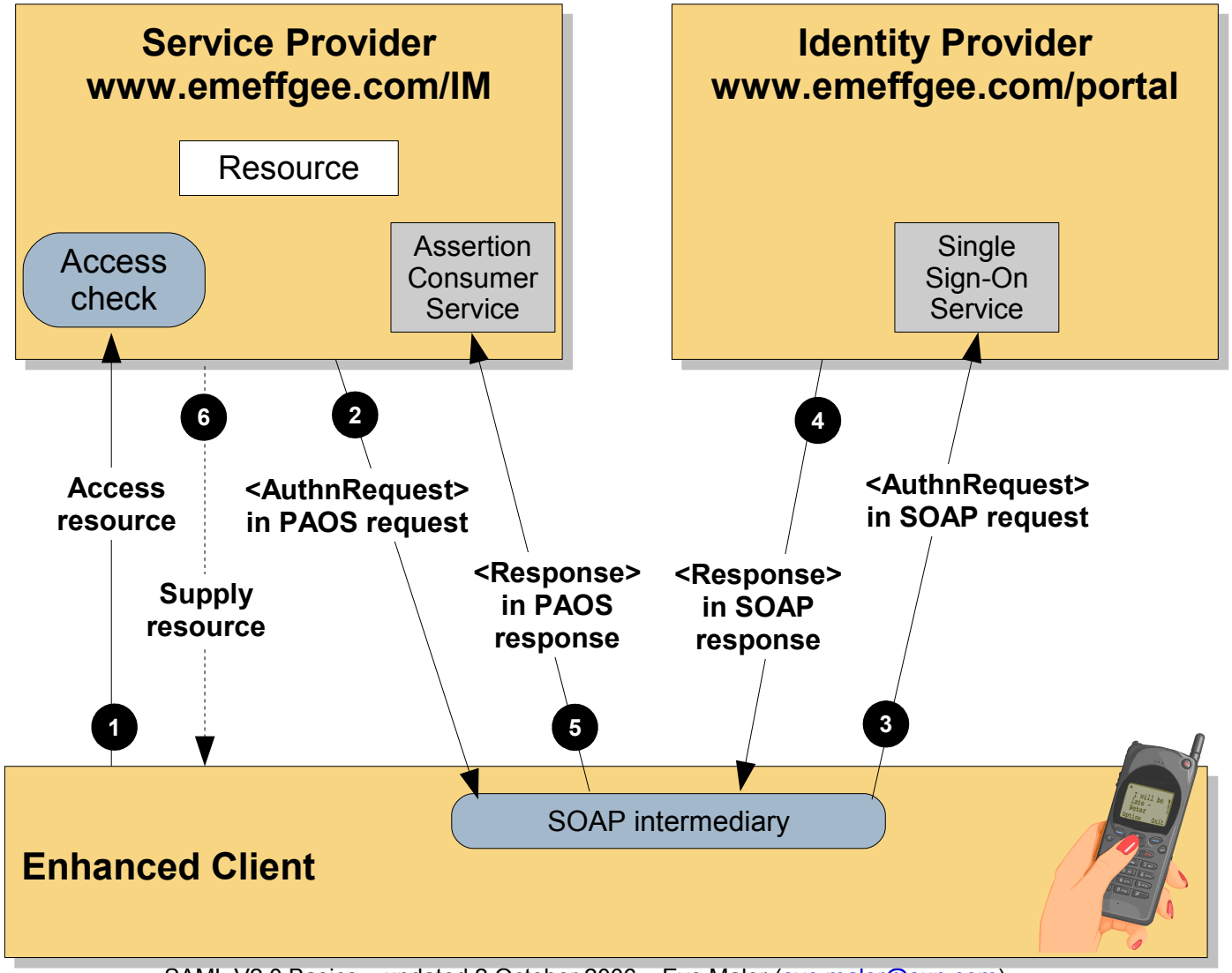


Enhanced client



Enhanced proxy

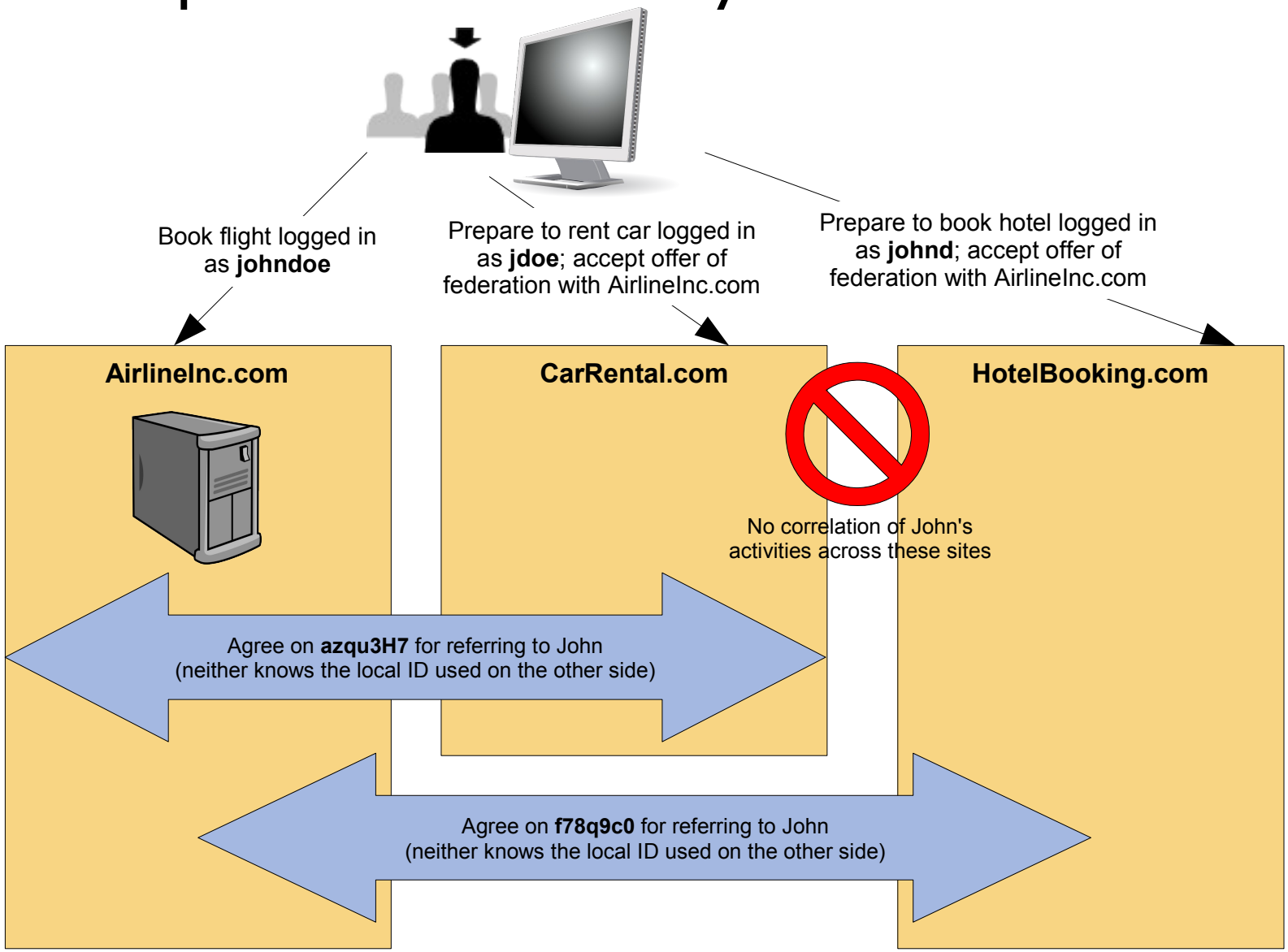
SSO using ECP



Account linking with privacy and flexibility

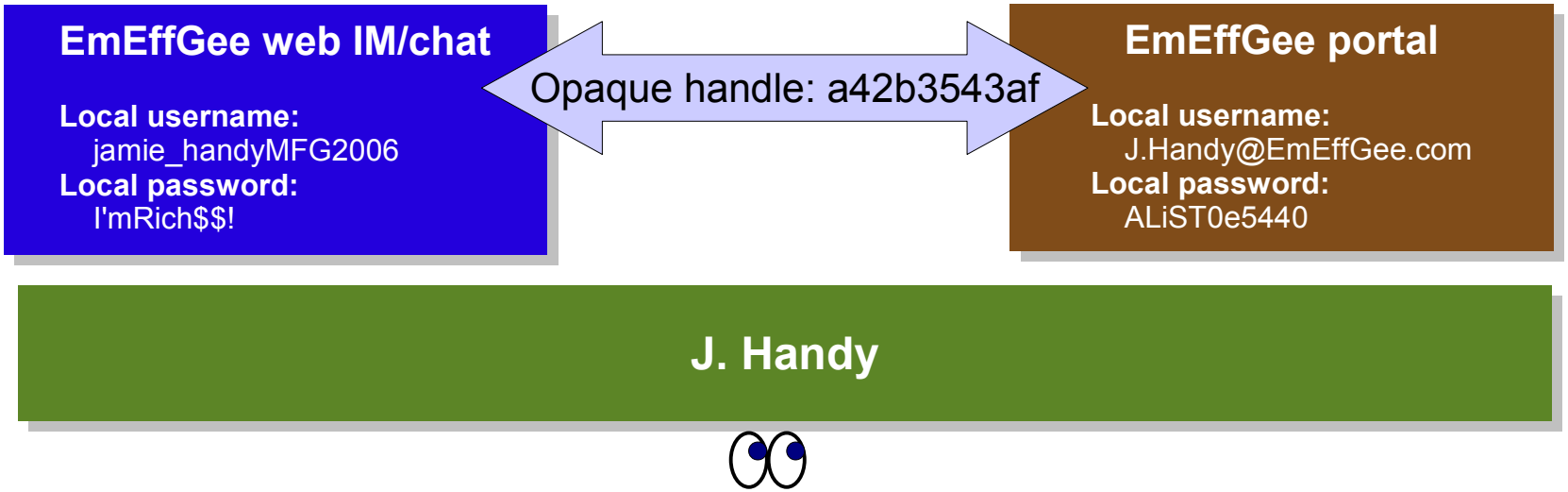
- SSO involves only one-way information flow
  - The IM/chat app need not have a "local account" for J. Handy at all
  - Typically, however, it does because its relationship with J. is non-trivial
- Two-way flow of information is often desired to synchronize identity data stores
- The two apps could become IdPs for each other by "opening the kimono" and sharing J.'s identifier for correlation (federation)
  - But it's J.'s kimono!

The basic use case for pairwise identity federation

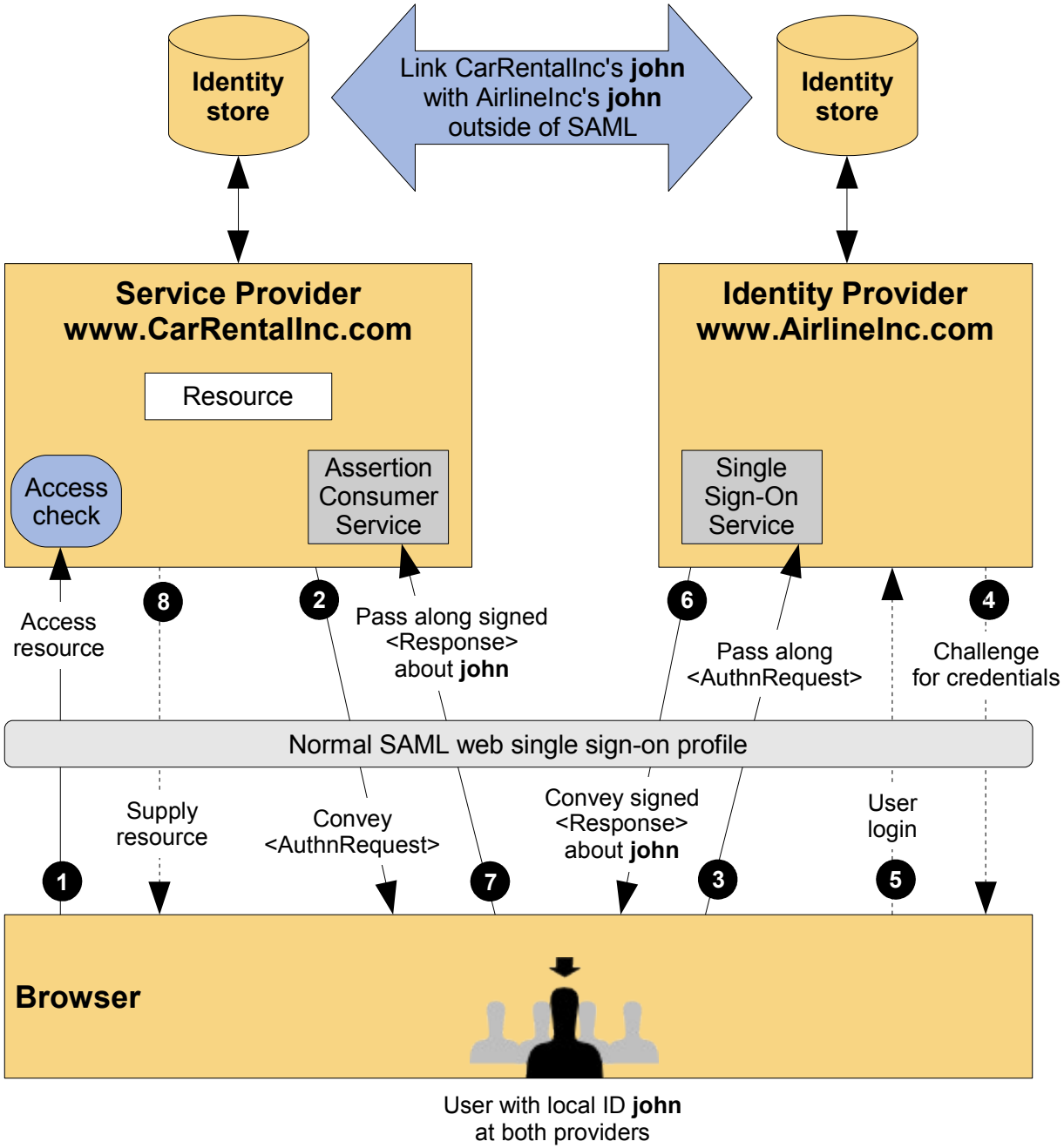


SAML's name identifier management profile

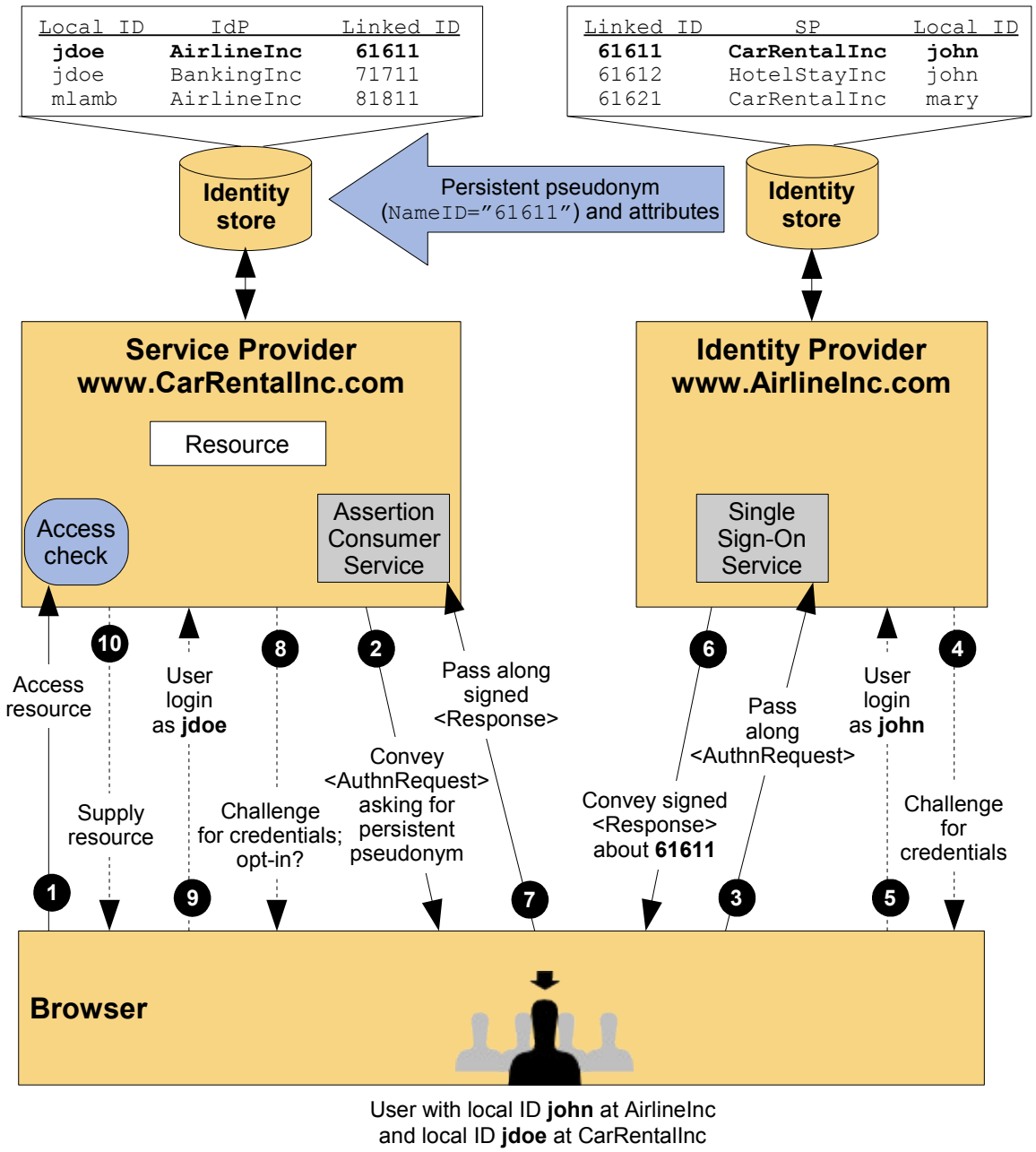
- Providers can set up pairwise-unique nicknames for J. Handy
  - One option is a **persistent pseudonym**, for an ongoing portal+IM app relationship
  - Another is a **transient pseudonym**, e.g. for single-session access granted to groups based on attributes



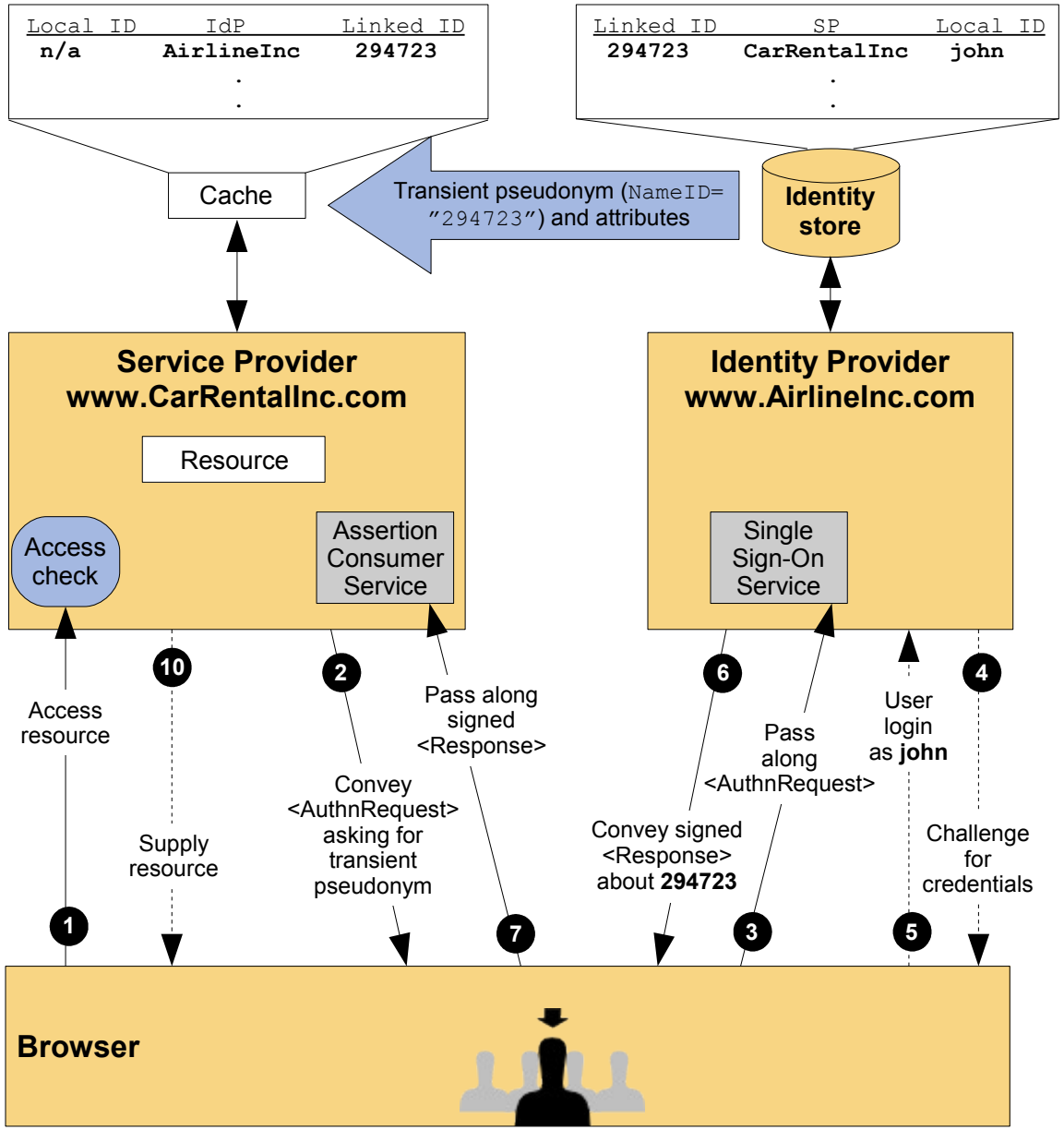
Out-of-band federation



Federation with a persistent pseudonym

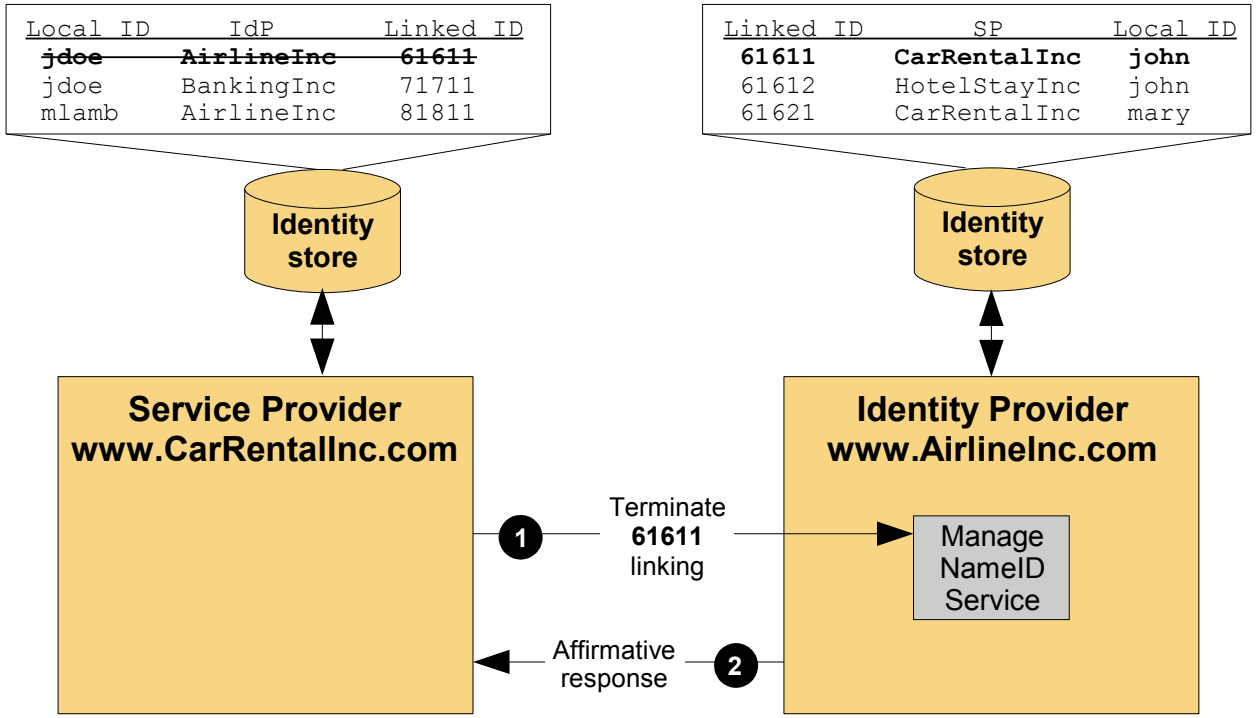


Federation with a transient pseudonym

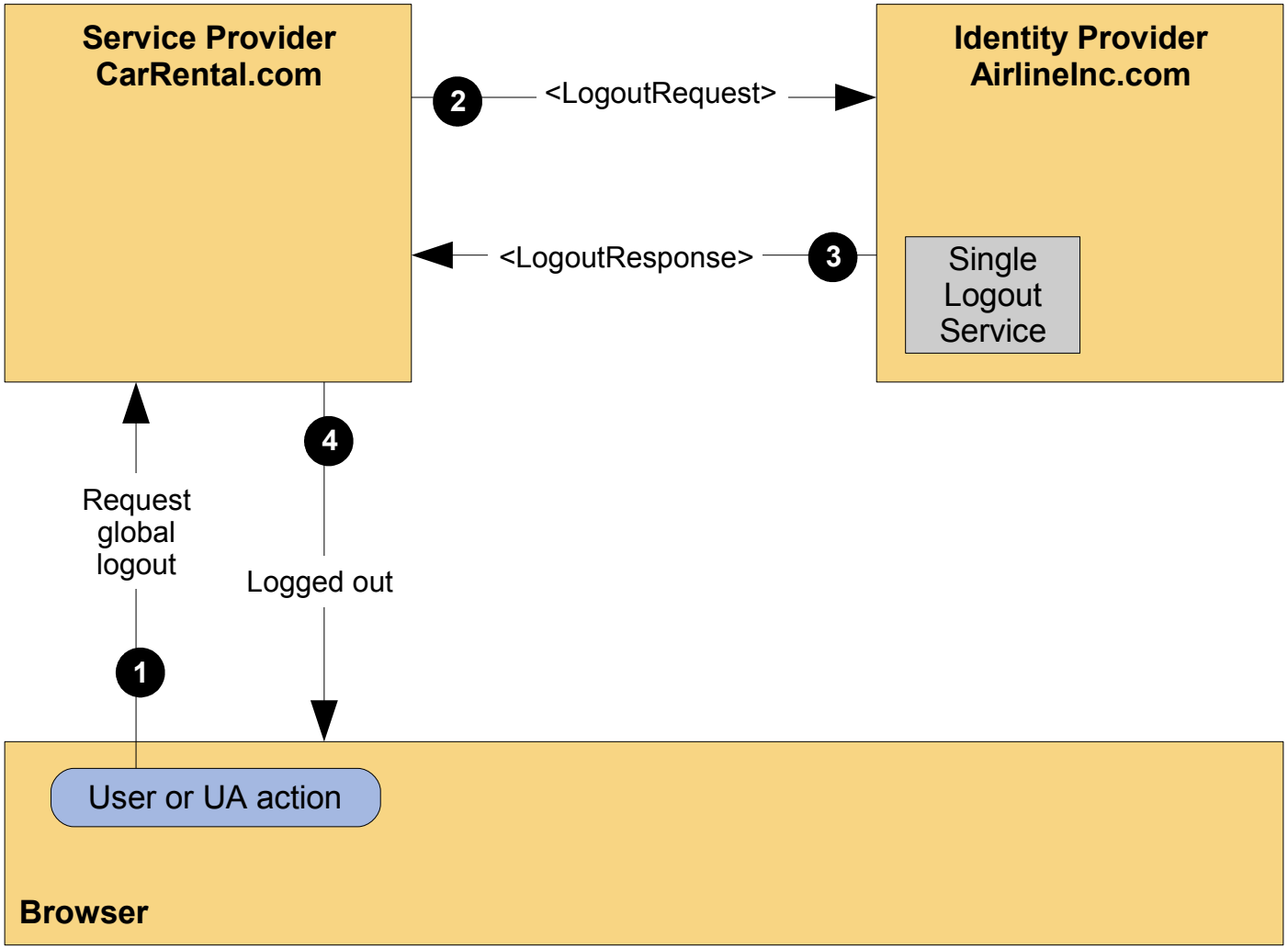


User with local ID **john** at AirlineInc

Federation termination



SP-initiated single logout



How you can get started

Development vs. deployment

- You shouldn't have to implement SAML support from scratch in applications
  - Open-source implementations for various languages and platforms
    - However, SAML V2.0 is still new in "roadmap" terms
    - OpenSAML is out in front on support
  - Free trials of products
    - E.g., Sun's Access Manager/Federation Manager are available for free through the "Red October" program
- Your lawyers and privacy advocates shouldn't have to start from scratch either
  - Use Liberty Alliance guidelines in building federation relationships

A real-world case study: Sun-BIPAC federation

- Sun provides an employee benefit: access to BIPAC, which provides insight on the U.S. political scene
  - BIPAC offers a web application for personalized information lookup by Congressional district
  - Some personalization is restricted by U.S. Law related to privacy
- Benefits of federation (N.B.: it uses Liberty ID-FF):
  - Cross-domain SSO from the Sun IdP to the BIPAC SP, in the course of which stronger authentication has been deployed
  - Privacy-enabled attribute exchange to allow anonymous – yet personalized – experiences
- See Enterprise Outsourcing paper for interesting deployment considerations and lessons

Resources

Some helpful resources

- SAML specs and outreach info:
  http://www.oasis-open.org/committees/security
- Liberty deployment guidelines:
  http://projectliberty.org/resources/guidelines.php
- SAML/Liberty Federation adoption info:
  http://projectliberty.org/about/marketadoption.php
- The IIW map in full resolution:
  http://photos.windley.com/albums/iiw2006a/IIW2006_identity_map
- Paper on Liberty Federation in Enterprise Outsourcing:
  http://www.idealliance.org/proceedings/xml05/abstracts/paper154.html
- Aggregation of many popular identity weblogs:
  http://www.planetidentity.org
- Some open-source projects involving SAML:
  http://OpenSSO.dev.java.net
  http://www.OpenSAML.org
  http://www.SourceID.org
  http://Lasso.Entrouvert.org
  http://ZXID.org

Any questions?

Thank you for your attention

Eve Maler
eve.maler@sun.com
Pushing String @ http://www.xmlgrrl.com/blog