



OpenID Connect Update

May 14, 2013

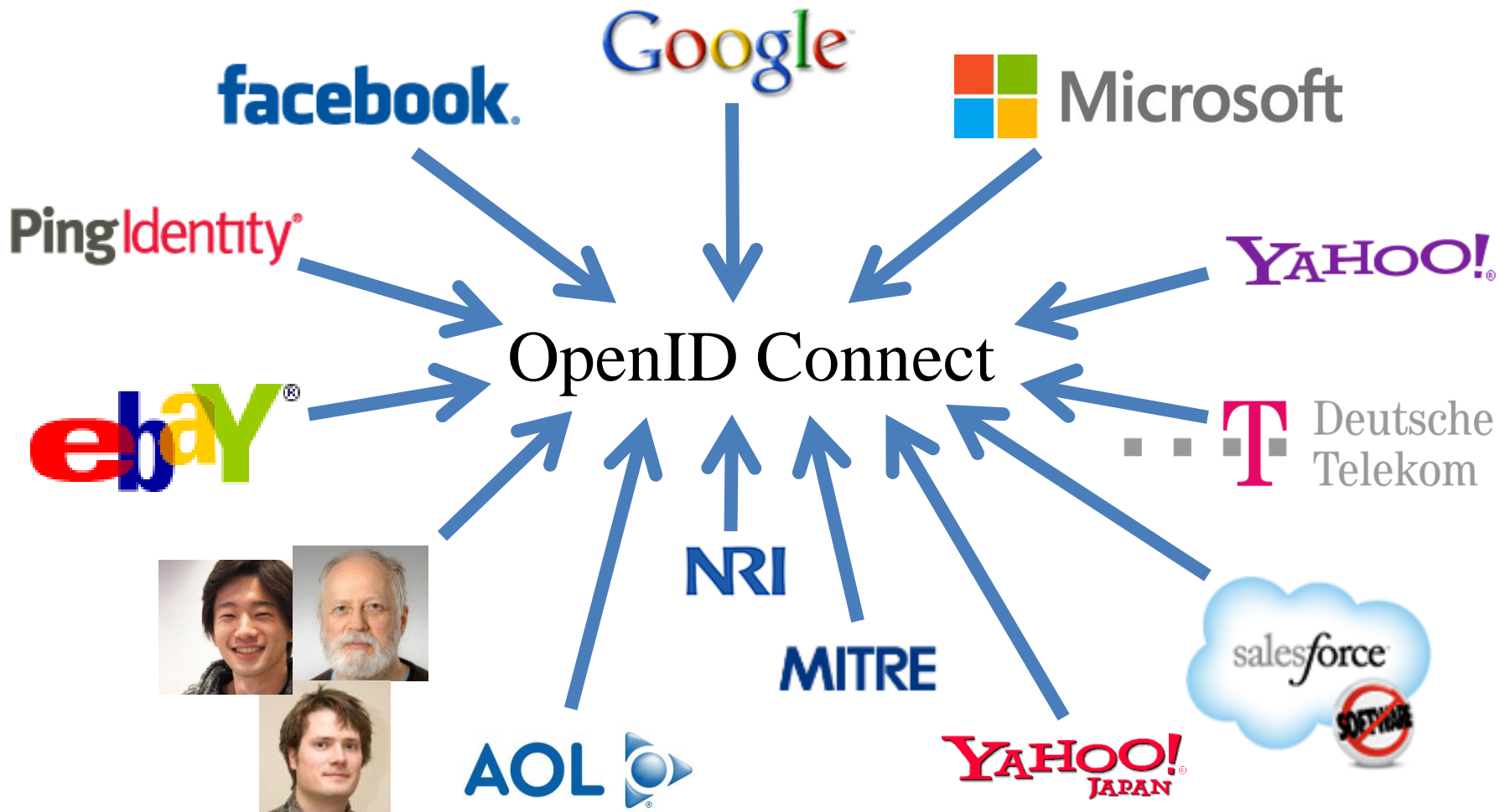
Dr. Michael B. Jones

Identity Standards Architect – Microsoft



OpenID

Working Together





OpenID Working Group Members

- Key working group participants:
 - Nat Sakimura – Nomura Research Institute – Japan
 - John Bradley – Ping Identity – Chile
 - Breno de Medeiros – Google – US
 - Axel Nennker – Deutsche Telekom – Germany
 - Torsten Lodderstedt – Deutsche Telekom – Germany
 - Roland Hedberg – Umeå University – Sweden
 - Andreas Åkre Solberg – UNINETT – Norway
 - Chuck Mortimore – Salesforce – US
 - George Fletcher – AOL – US
 - Justin Richer – Mitre – US
 - Nov Mataka – Independent – Japan
 - Mike Jones – Microsoft – US
- *By no means an exhaustive list!*



OpenID

OpenID Connect Intro

- Simple identity layer on top of OAuth 2.0
- Enables clients to verify identity of end-user
- Enables clients to obtain basic profile info
- REST/JSON interfaces → low barrier to entry



OpenID OpenID Connect Range

- Spans use cases, scenarios
 - Internet, Enterprise, Mobile, Cloud
- Spans security & privacy requirements
 - From non-sensitive information to highly secure
- Spans sophistication of claims usage
 - From basic default claims to specific requested claims to aggregated and distributed claims
- Maximizes simplicity of implementations
 - Uses existing IETF specs: OAuth 2.0, JWT, etc.
 - Lets you build only the pieces you need



OpenID Key Diffs from OpenID 2.0

- Support for native client applications
- Identifiers using e-mail address format
- UserInfo endpoint for simple claims about user
- Designed to work well on mobile phones
- Uses JSON/REST, rather than XML
- Support for encryption and higher LOAs
- Support for distributed and aggregated claims
- Support for session management, including logout
- Support for self-issued identity providers



OpenID Presentation Overview

- Introduction
- Design Philosophy
- A Look Under the Covers
- Overview of Connect Specs
- Underpinnings
- Timeline
- Open Issues
- Next Steps
- Resources



Design Philosophy

Simple Things Simple

Complex Things Possible



OpenID

Simple Things Simple

UserInfo endpoint for
simple claims about user

Designed to work well on
mobile phones



OpenID

How We Make It Simple

- Build on OAuth 2.0
- Use JavaScript Object Notation (JSON)
- Build only the pieces that you need
- *Goal: Easy implementation on all modern development platforms*



OpenID

Complex Things Possible

Encrypted Claims

Aggregated Claims

Distributed Claims



OpenID Connect Interop Status

- About to begin 5th round of interop testing
- Interop data at <http://osis.idcommons.net/>
- By the numbers:
 - 13 implementations participating
 - 66 cross-solution test results recorded
 - 110 feature tests defined
 - 1,260 feature test results recorded
 - 93 members of interop mailing list
 - 752 messages to interop mailing list



OpenID A Look Under the Covers

- ID Token
- Claims Requests
- UserInfo Claims
- Example Protocol Messages



ID Token

- JWT representing logged-in session
- Claims:
 - `iss` – Issuer
 - `sub` – Identifier for subject (user)
 - `aud` – Audience for ID Token
 - `iat` – Time token was issued
 - `exp` – Expiration time
 - `nonce` – Mitigates replay attacks



OpenID ID Token Claims Example

```
{  
  "iss": "https://server.example.com",  
  "sub": "248289761001",  
  "aud": "0acf77d4-b486-4c99-bd76-074ed6a64ddf",  
  "iat": 1311280970,  
  "exp": 1311281970,  
  "nonce": "n-0S6_WzA2Mj"  
}
```



Claims Requests

- Basic requests made using OAuth scopes:
 - `openid` – Declares request is for OpenID Connect
 - `profile` – Requests default profile info
 - `email` – Requests email address & verification status
 - `address` – Requests postal address
 - `phone` – Requests phone number & verification status
 - `offline_access` – Requests Refresh Token issuance
- Requests for individual claims can be made using JSON “claims” request parameter



UserInfo Claims

- sub
- name
- given_name
- family_name
- middle_name
- nickname
- preferred_username
- profile
- picture
- website
- gender
- birthdate
- locale
- zoneinfo
- updated_at
- email
- email_verified
- phone_number
- phone_number_verified
- address



OpenID

UserInfo Claims Example

```
{  
  "sub": "248289761001",  
  "name": "Jane Doe",  
  "given_name": "Jane",  
  "family_name": "Doe",  
  "email": "janedoe@example.com",  
  "email_verified": true,  
  "picture": "http://example.com/janedoe/me.jpg"  
}
```



Authorization Request Example

```
https://server.example.com/authorize  
?response_type=token%20id_token  
&client_id=0acf77d4-b486-4c99-bd76-074ed6a64ddf  
&redirect_uri=https%3A%2F%2Fclient.example.com%2Fcb  
&scope=openid%20profile  
&state=af0ifjslkdj  
&nonce=n-0S6_WzA2Mj
```



Authorization Response Example

HTTP/1.1 302 Found

Location: <https://client.example.com/cb>

#access_token=mF_9.B5f-4.1JqM

&token_type=bearer

&id_token=eyJhbGZlNiJ9.eyJz9Glnw9J.F9-V4IvQ0Z

&expires_in=3600

&state=af0ifjsldkj



OpenID UserInfo Request Example

```
GET /userinfo?schema=openid HTTP/1.1  
Host: server.example.com  
Authorization: Bearer mF_9.B5f-4.1JqM
```



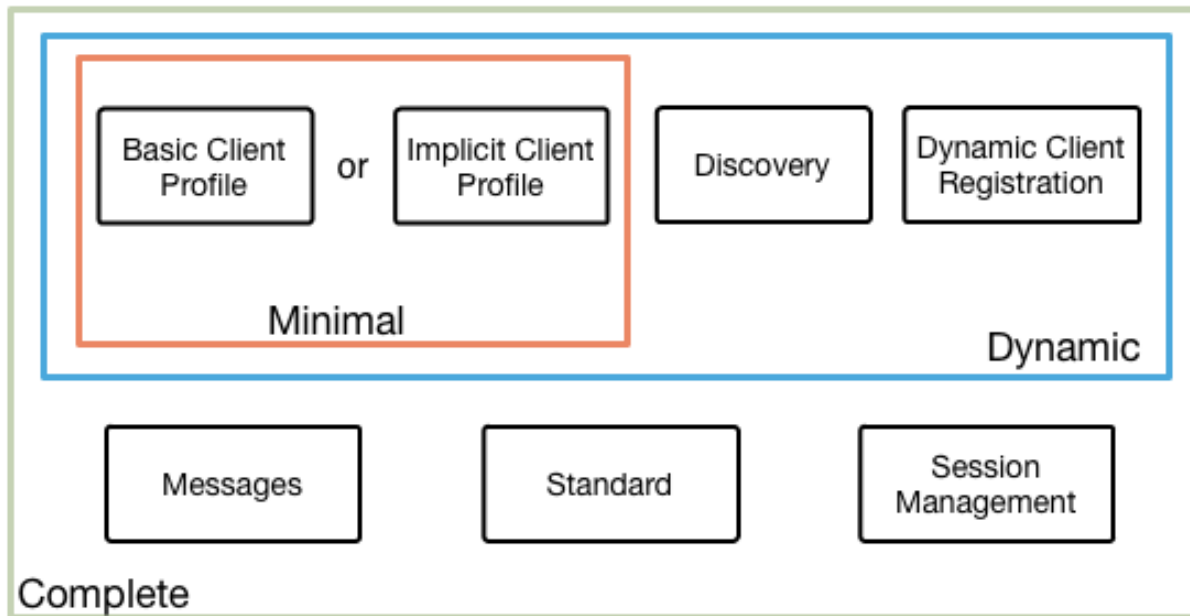
OpenID

Connect Specs Overview

20 March 2013

<http://openid.net/connect>

OpenID Connect Protocol Suite



Underpinnings





Basic Client Profile

- Single, simple, self-contained Web client spec
 - For clients using OAuth “code” flow
- All you need for Web server-based RP
 - Using pre-configured set of OPs
- http://openid.net/specs/openid-connect-basic-1_0.html



OpenID

Implicit Client Profile

- Single, simple, self-contained Web client spec
 - For clients using OAuth “implicit” flow
- All you need for user agent-based RPs
 - Using pre-configured set of OPs
- http://openid.net/specs/openid-connect-implicit-1_0.html



OpenID Discovery & Registration

- Enables dynamic configurations in which sets of OPs and RPs are not pre-configured
 - Necessary for *open* deployments
- Discovery enables RPs to learn about OP endpoints
- Dynamic registration enables RPs to use OPs they don't have pre-existing relationships with
- http://openid.net/specs/openid-connect-discovery-1_0.html
- http://openid.net/specs/openid-connect-registration-1_0.html



OpenID

Messages & Standard

- Messages spec defines data formats exchanged in OpenID Connect messages
- Standard spec is HTTP binding for Messages
 - (Basic and Implicit are profiles of Messages and Standard)
- Needed for OPs, native client apps, and RPs needing functionality not in Basic
 - E.g., requesting claims not in default UserInfo set
- http://openid.net/specs/openid-connect-messages-1_0.html
- http://openid.net/specs/openid-connect-standard-1_0.html



OpenID

Session Management

- For OPs and RPs needing session management capabilities
 - Enables logout functionality
 - Enables account switching
- http://openid.net/specs/openid-connect-session-1_0.html



OpenID OAuth Response Types

- Defines and registers additional OAuth response types:
 - `id_token`
 - `none`
- And also defines and registers combinations of `code`, `token`, and `id_token` response types
- http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html



Underpinnings

- OAuth 2.0 family of specs
 - OAuth 2.0 Core – RFC 6749
 - OAuth 2.0 Bearer – RFC 6750
 - OAuth 2.0 Assertions
 - OAuth 2.0 JWT Assertions Profile
- JWT and JOSE family of specs
 - JSON Web Token (JWT)
 - JSON Web Signature (JWS)
 - JSON Web Encryption (JWE)
 - JSON Web Algorithms (JWA)
 - JSON Web Key (JWK)
- WebFinger discovery spec



Timeline

- Artifact Binding working group formed, Mar 2010
- Major design issues closed at IIW, May 2011
 - Result branded “OpenID Connect”, May 2011
- Functionally complete specs, Jul 2011
- 2nd interop testing round, Sep-Nov 2011
- Simpler specs incorporating dev feedback, Oct 2011
- Published First Implementer’s Drafts, Dec 2011
- 3rd interop testing round, Feb 2012 to May 2012
- ***OpenID Connect won Best Innovation/New Standard award at EIC, April 2012***
- Revised specs incorporating more feedback, June 2012
- 4th interop testing round, June 2012 to May 2013
- Second Implementers Drafts to be published, May 2013
- 5th interop testing round to begin, May 2013
- Working on stabilizing JOSE encryption spec (JWE), ongoing



Risks to Timely Completion

- Dependencies on IETF specs/processes
 - OAuth specifications:
 - JWT, OAuth Assertions, OAuth JWT Assertions, OAuth Dynamic Registration
 - JOSE specifications:
 - JWS, JWE, JWA, JWK
 - Discovery-related specifications:
 - WebFinger, acct URI
- IETF could change/delay any of these



OpenID

Primary IETF Open Issue

- draft-ietf-jose-json-web-encryption (JWE)
 - Some WG members still trying to change things
 - Stabilizing JWE the main dependency for finishing OpenID Connect
- Less risk of change for other dependencies



Next Steps

- Update specs for recent issue resolutions
 - Minor updates to both JOSE and Connect specs
- Publish new proposed implementer's drafts
 - Not yet final because of IETF spec dependencies
 - Membership vote to approve them
- Create 5th OpenID Connect Interop (OC5)
- ***Continued deployment and feedback***
- Make determination that IETF dependencies stable
- Publish final specification drafts
 - Once dependencies are resolved
 - Membership vote to approve final specifications



Resources

- OpenID Connect
 - <http://openid.net/connect/>
- OpenID Connect Working Group Mailing List
 - <http://lists.openid.net/mailman/listinfo/openid-specs-ab>
- OpenID Connect Interop Wiki
 - <http://osis.idcommons.net/>
- OpenID Connect Interop Mailing List
 - <http://groups.google.com/group/openid-connect-interop>
- Mike Jones' Blog
 - <http://self-issued.info/>
- Nat Sakimura's Blog
 - <http://nat.sakimura.org/>
- John Bradley's Blog
 - <http://www.thread-safe.com/>

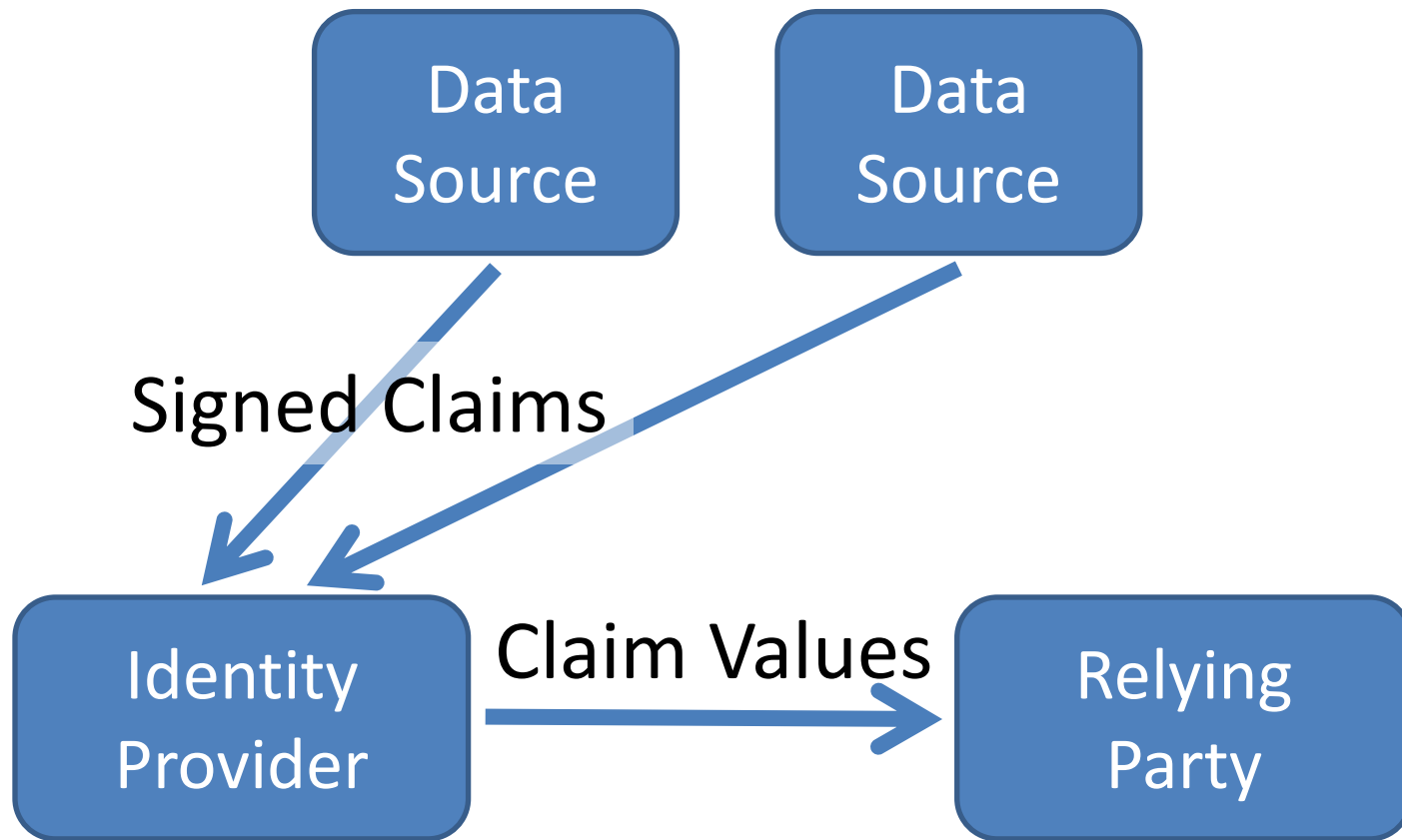


BACKUP SLIDES



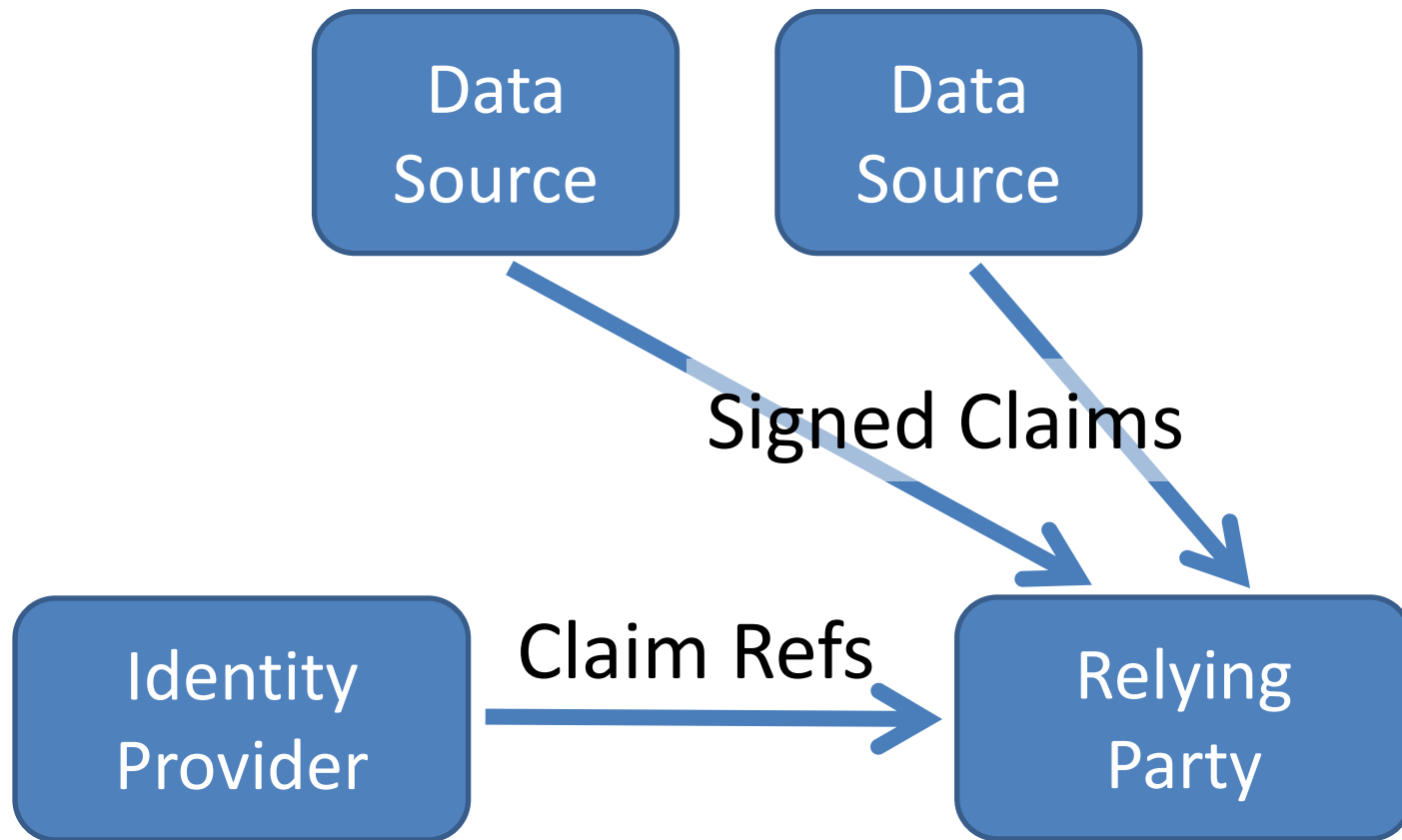
OpenID

Aggregated Claims





Distributed Claims





OpenID

Connect OAuth Specs

- draft-ietf-oauth-v2
 - Now RFC 6749
- draft-ietf-oauth-v2-bearer
 - Now RFC 6750
- draft-ietf-oauth-urn-sub-ns
 - Now RFC 6755
- draft-ietf-oauth-v2-threatmodel
 - Now RFC 6819
- draft-ietf-oauth-assertions
 - Ready for WGLC
- draft-ietf-oauth-json-web-token
 - Ready for WGLC
- draft-ietf-oauth-oauth-jwt-bearer
 - Ready for WGLC
- draft-ietf-oauth-dyn-reg
 - In WGLC



OpenID

Connect JOSE Specs

- draft-ietf-jose-json-web-signature
 - Ready for WGLC
- draft-ietf-jose-json-web-encryption
 - Ready for WGLC
- draft-ietf-jose-json-web-algorithms
 - Ready for WGLC
- draft-ietf-jose-json-web-key
 - Ready for WGLC



OpenID Connect Apps Area Specs

- draft-ietf-appsawg-webfinger
 - In IESG review
- draft-ietf-appsawg-acct-uri
 - In IESG review



Developer Feedback Incorporated

- Ask: Simpler, more modular specs
 - Created Basic Client Profile and Implicit Client Profile
 - Messages and Standard also simplified
- Ask: Enable single-sign-on without using UserInfo
 - Can now receive just an ID Token, if desired
- Ask: Remove Check ID endpoint
 - ID Token validation now done directly by RP
- Ask: Self-issued identity providers
 - Self-issued OP mechanisms defined
- *Not a comprehensive list of feedback incorporated*