SOR-Registry Strawman ID Match API

About ID Match

ID Match refers to the task of determining if a person presented by a system of record is already known to the Identity Management System.

There are several characteristics of an ID Match implementation.

Coordinated vs Independent Attributes

In a **coordinated** implementation, all systems of record agree to a single set of "golden" attributes, and the ID Match component is authoritative for these attributes. When an SoR presents attributes for matching, the ID Match component matches against the single, golden set of attributes.

In an **independent** implementation, each system of record is authoritative for all of its own attributes. When an SoR presents attributes for matching, the ID Match component may match against a canonical representation of these attributes as managed by the IdMS, but it may also match against each SoR's representation of these attributes.

Synchronous vs Asynchronous Match Resolution

When a partial/potential match occurs, the client may be able to resolve the match **synchronously** by presenting information to the data entry personnel and submitting a forced resolution request.

If the client is not able to do so, then the match is resolved **asynchronously**, usually by a notification going to a match administrator, who can then view the potential match and resolve it.

When matching is performed via batch operations, it must be asynchronous.

Implementation at Registry vs Standalone

ID Match can be performed at the **Person Registry** or as a **standalone** service.

Match Before Registry vs Match At Registry

ID Match can be performed **before** a record is added to the Person Registry, or **at** the Person Registry. In either case, the ID Match might be performed by the Person Registry, or by a standalone service.

Algorithm

The matching algorithm is generally out of scope for the ID Match API.

About the ID Match API

This API is used to obtain a unique reference to a Person known to the IdMS based on data known to an SOR. The Person may or may not be known to the IdMS at the time of the query, but generally the Person will be new to the SOR.

This API is exposed by the ID Match component. The Registry component may also expose this API, either because it implements ID Match natively or because it brokers requests to the ID Match component.

This draft is based on the ID Match Strawman. All services are mandatory unless otherwise stated.

⚠ Identifiers used by the Match API may be returned in either string or number notation. This includes the match reference identifier, the match request identifier, and any local identifiers.

People

The ID Match API operates with the primary resource being People. The goal of a client is to obtain a reference identifier for a person.

Reference Identifier Request

Obtain a unique reference identifier from the ID Match component in order to canonically identify the person presented by the SOR.

Any attributes defined in the SOR-Registry Core Schema Specification are permitted in ID Match requests. However, it is the following list is likely to represent the most common attributes:

- identifier/sor (sorId, provided in the URL)
- address
- affiliation
- dateOfBirth
- emailAddress
- gender
- identifier/national (may be hashed)
- identifier/network
- identifier/enterprise
- name
- telephoneNumber

The specific sorAttributes that are required may vary by instance.

Note: The reference identifier returned to an SOR may match an identifier previously known to the SOR. For example, if an employee returns after an absence of several years, the IdMS may have the original reference identifier whereas the HRMS may have purged its copy.

Request

In this example, *sis* is the SOR label and *971194843* is the SOR ID for the candidate.

⚠ If the SOR ID is sensitive (such as being an SSN), placing it in the URL may not be ideal. In such a case, use of a salted-hashed identifier is recommended.

```
PUT /v1/people/sis/971194843
{
  "sorAttributes":
  {
    "names":[
      {
        "type":"official",
        "given":"Pat",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
    "identifiers":[
      {
        "type":"national",
        "identifier":"3B902AE12DF55196"
      }
    ],
    "telephoneNumbers":[
      {
        "type":"mobile",
        "number":"8185551234"
      }
    ]
  }
}
```

Request: Search Only

A *search only* request is similar to a regular request, except that a new identity will never be created as a result of the request. Note that a search only request does *not* imply that the requesting SOR intends to add a role record for an identity, and so is *not* an instruction for the match engine to change any state (ie: it is read-only from the match engine perspective).

```
GET /v1/people/sis/971194843?names.0.type=official&names.0.given=Pat&names.0.family=Lee&dateOfBirth=1983-03-
18&identifiers.0.type=national&identifiers.0.identifier=3B902AE12DF55196&telephoneNumbers.0.
type=mobile&telephoneNumbers.0.number=8185551234
```

The same search only request may be submitted via `POST` to avoid embedding sensitive information in `GET` URLs.

```
POST /v1/people/sis/971194843
{
  "sorAttributes":
  {
    "names":[
      {
        "given":"Patricia",
        "family":"Lee",
        "type":"official"
      }
    ],
    "dateOfBirth":"1983-03-18",
    "identifiers":[
      {
        "type":"national",
        "identifier":"3B902AE12DF55196"
      }
    ],
    "telephoneNumbers":[
      {
        "type":"mobile",
        "identifier":"8185551234"
      }
    ]
  }
}
```

Response: Unique Match Found

An existing identity was found matching the specified attributes.

```
200 OK
{
  "referenceId":"M225127891"
}
```

Response: New Identifier Assigned

A new identity was created, since no existing identity matched the specified attributes. *New Identifier Assigned* and *Unique Match Found* are effectively equivalent from the perspective of the client. This response will never be returned from a search only request.

```
201 Created
{
  "referenceId":"M225127891"
}
```

Response: No Match Found

This response is only valid as a result of a search only request.

```
404 Not Found
```

Response: With Additional Identifiers

Implementations may return additional identifiers in a successful (200 or 201) response.

```
200 OK
{
  "referenceId":"M225127891",
  "identifiers":[
    {
      "type":"network",
      "identifier":"pl388"
    },
    {
      "type":"enterprise",
      "identifier":"905003148"
    }
  ]
}
```

Response: With Golden Attributes

Coordinated implementations may return golden record attributes in a successful (200 or 201) response.

```
200 OK
{
  "referenceId":"M225127891",
  "identifiers":[
    {
      "type":"network",
      "identifier":"pl388"
    },
    {
      "type":"enterprise",
      "identifier":"905003148"
    }
  ],
  "golden":{
    "names":[
      {
        "type":"official",
        "given":"Patricia",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
  }
}
```

Response: Potential Match Found

This response allows an SOR to interactively select from potential matches. (See also Forced Reconciliation Request, below.) This response may include as few as one candidate. Implementation of this response is optional if the *Externally Handled* response is implemented.

confidence is an integer from 0 through 100, where 100 indicates maximum confidence. It is optional, but strongly recommended to provide assistance to the administrator trying to resolve the potential match.

Attributes may be returned to facilitate the selection of a candidate. The attributes may include any attribute specified in the SOR-Registry Core Schema Strawman, even attributes that the matching engine does not use to perform matching. The specific set of attributes returned may vary by local implementation.

To indicate a new person, the Identity Match engine may create a provisional record returned in the `300 Multiple Choices` response, with a suitable referenceId assigned. Alternately, the referenceId may be set to `new`. Either way, the original data submitted is provided as a candidate with *confidence* omitted.

The Identity Match engine may elect to assign a *match request ID* to the transaction that generated the `300 Multiple Choices` response. The match request ID is optional. If assigned, it is the same identifier as used to retrieve Pending Matches, and may be used in that context. If assigned, the match request ID *must* be provided when making a Forced Reconciliation Request to resolve the `300 Multiple Choices` response.

```
300 Multiple Choices
{
  "matchRequest":"1009",
  "candidates":
  [
    {
      "confidence":"85",
      "referenceId":"M219488003",
      "attributes":
      [
        {
          "sor":"HRMS",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"089010023"
            },
            {
              "type":"network",
              "identifier":"pl292"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Patricia",
              "family":"Lee"
            }
          ],
          "ou":"Biomedical Informatics"
        },
        {
          "sor":"Alumni",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"A330-200"
            },
            {
              "type":"network",
              "identifier":"pl292"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Patricia",
              "family":"Lee"
            }
          ],
          "ou":"Class of 1997"
        }
      ],
      "identifiers":[
        {
          "type":"network",
          "identifier":"pl292"
        },
        {
          "type":"enterprise",
          "identifier":"905008772"
        }
      ]
    },
    {
      "confidence":"71",
      "referenceId":"M523441767",
      "attributes":
      [
        {
          "sor":"guest",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"pl388"
            },
            {
              "type":"network",
              "identifier":"pl388"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Patricia",
              "family":"Lee"
            }
          ],
          "telephoneNumbers":[
            {
              "type":"mobile",
              "number":"8185551234"
            }
          ]
        }
      ],
      "identifiers":[
        {
          "type":"network",
          "identifier":"pl388"
        },
        {
          "type":"enterprise",
          "identifier":"905003148"
        }
      ]
    },
    {
      "referenceId":"new",
      "attributes":
      [
        {
          "sor":"SIS",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"971194843"
            }
          ]
          "names":[
            {
              "type":"official",
              "given":"Pat",
              "family":"Lee"
            }
          ],
          "telephoneNumbers":[
            {
              "type":"mobile",
              "number":"8185551234"
            }
          ]
        }
      ]
    }
  ]
}
```

In a coordinated implementation, golden record attributes may be included for each candidate:

```
300 Multiple Choices
{
  "matchRequest":"1009",
  "candidates":
  [
    {
      "confidence":"85",
      "referenceId":"M219488003",
      "golden":{
        "names":[
          {
            "type":"official",
            "given":"Patricia",
            "family":"Lee"
          }
        ],
        "dateOfBirth":"1983-03-18",
      },
      "attributes":
      [
       ...
      ]
    }
  ]
}
```

The ID Match engine may include a human readable explanation as to why a candidate was returned. This explanation is intended to provide guidance when an administrator is trying to resolve a potential match situation.

```
300 Multiple Choices
{
  "matchRequest":"1009",
  "candidates":
  [
    {
      "confidence":"85",
      "referenceId":"M219488003",
      "explanation":"Family name exact match, given name initial match",
      "attributes":
      [
       ...
      ]
    }
  ]
}
```

Response: Potential Match Found (Externally Handled)

Some configurations may not support an interactive transaction over the API to resolve a conflict. In this sort of scenario, the ID Match engine will submit a reconciliation request to a Reconciliation Manager (probably via email with a URL for resolution). The engine is responsible for maintaining enough state to be able to handle a manual reconciliation at a later (but not too much later) time. This response will never be returned from a search only request.

See also Console Support Operations, below.

```
202 Accepted
```

An optional match request identifier may be returned as part of this response.

```
202 Accepted
{
  "matchRequest":"1009"
}
```

Response: Error With Request

Errors could include failing to submit required attributes, submitting invalid values for attributes, etc.

```
400 Bad Request
```

An optional human readable error message may be included in the response body.

```
400 Bad Request
{
  "error":"Required field not provided"
}
```

Forced Reconciliation Request

After a `300 Multiple Choices` response, the client must submit a modified request with an ID Match Reference Identifier to link to an existing record, or an indication that a new person should be created (by passing new as the `referenceId`. This will generally result in `200 OK` or `201 Created`. Only candidates listed in the `300` response should be included in the forced reconciliation request, however it is up to an individual implementation as to whether or not to accept an ID Match Reference Identifier provided in a given Forced Reconciliation Request.

The Identity Match Engine may indicate a forced reconciliation request is being attempted using out of date information by returning 409 Conflict.

```
PUT /v1/people/sis/971194843
{
  "sorAttributes":
  {
    "names":[
      {
        "type":"official",
        "given":"Pat",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
    "identifiers":[
      {
        "type":"national",
        "identifier":"3B902AE12DF55196"
      }
    ],
    "telephoneNumbers":[
      {
        "type":"mobile",
        "number":"8185551234"
      }
    ]
  },
  "referenceId":"M523441767"
}
```

If a Match Request Identifier was returned in a `202` or `300` response, it must be included in the Forced Reconciliation Request.

```
PUT /v1/people/sis/971194843
{
  "matchRequest":"1009",
  "sorAttributes":
  {
    ...
  },
  "referenceId":"M523441767"
}
```

Update Match Attributes

The Identity Match engine must be kept up to date if attributes used for matching are updated. (For example, if a person's name changes, the SOR must update the name attributes in case the person subsequently shows up via a different SOR.) This can also be handled out of band, eg by sync'ing the match database with a registry database, or by the two databases being the same.

A complete set of attributes are provided to replace the existing set. The Identity Match engine may elect to keep replaced attributes for fuzzy matching against historical records, or for other purposes.

For a *coordinated* implementation, the reference identifier must be included as the Identity Match engine need not maintain a full set of per-SOR attributes. For an *independent* implementation, the reference identifier is optional.

Operation	Out of Band Sync	Coordinated	Independent
Update Match Request	Do Not Implement	Required, including reference identifier	Required (reference identifier optional)
Request Inventory of Requests	Optional	Required	Required
Request Current Values	Optional	Required, including reference identifier	Required (reference identifier optional)
Delete Current Values	Do Not Implement	Do Not Implement	Required

Request

Typically, there is a Reference Identifier associated with the SOR and SORID. This request is therefore a request to update the System of Record attributes for use in future match requests, and the match engine may simply update its internal state and reply appropriately. Requesting a rematch of an already matched record is not supported via this mechanism.

If the requested SOR and SORID represents a pending match request (ie: there is no Reference ID associated with the request), the match engine should treat the request as a standard match request, but using the updated attributes. The request is most likely a SOR submitting updated attributes to fix a prior request that resulted in a pending match.

```
PUT /v1/people/sis/971194843
{
  "sorAttributes":
  {
    "names":[
      {
        "type":"official",
        "given":"Patricia",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
    "identifiers":[
      {
        "type":"national",
        "identifier":"3B902AE12DF55196"
      },
      {
        "type":"referenceId",
        "identifier":"M523441767"
      }
    ],
    "telephoneNumbers":[
      {
        "type":"mobile",
        "number":"8185551234"
      }
    ]
  }
}

200 OK
```

Request Inventory of Requests

The list of all requests made within an SOR may be obtained. The response is a list of SORIDs.

```
GET /v1/people/sis

200 OK
{
  "sorids":
  [
    "971194843",
    "980121418",
    "937762041",
    "915233810"
  ]
}
```

Request Current Values

The current values of match attributes as known to the match engine may be retrieved by the SOR. Note that if a match has previously been made, the `referenceId` will be included in the response. (If the request is pending manual resolution, the `referenceId` attribute will be omitted.) The match engine may elect to return former values for attributes, so long as they are suitably typed (eg: `fka` for a name).

```
GET /v1/people/sis/971194843

200 OK
{
  "sorAttributes":
  {
    "names":[
      {
        "type":"official",
        "given":"Patricia",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
    "identifiers":[
      {
        "type":"national",
        "identifier":"3B902AE12DF55196"
      }
    ],
    "telephoneNumbers":[
      {
        "type":"mobile",
        "number":"8185551234"
      }
    ]
  },
  "requestTime":"2013-06-08T11:23:37Z",
  "resolutionTime":"2013-06-10T11:23:37Z",
  "referenceId":"M523441767"}
```

This operation may also return `403 Unauthorized` or `404 Not Found`.

In a coordinated implementation, this operation should return the attributes as provided by the specified SOR, *not* the golden attributes. To obtain the golden attributes, use a Reference Identifier Obtain SOR Records request. If the SOR attributes are not tracked, then the `sorAttributes` entry should be omitted.

Delete Current Values

The match attributes for the SOR can be deleted. Typically, this should only be done when a record was added erroneously, as the match engine will perform better if it has access to historical records.

```
DELETE /v1/people/sis/971194843

200 OK
```

This operation may also return `403 Unauthorized` or `404 Not Found`.

Pending Matches

In order to support an Identity Console, certain additional resources are needed. These operations are mandatory, in order to support decoupling of services (in this case the Identity Match engine from the Identity Console). However where a given implementation implements interactive, stateless resolution (ie: it returns `300 Multiple Choices` and not `202 Accepted` for potential matches), the concept of "pending matches" may not apply, and these operations may effectively be no-ops.

Note that resolving a pending reconciliation request is handled via the Forced Reconciliation Request, described above.

Request Pending Matches

```
GET /v1/matchRequests?status=pending

200 OK
{
  "matchRequests":
  {
    "1009":
    {
      "attributes":
        {
          "sor":"SIS",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"971194843"
            }
          ]
          "names":[
            {
              "type":"official",
              "given":"Pat",
              "family":"Lee"
            }
          ],
          "telephoneNumbers":[
            {
              "type":"mobile",
              "number":"8185551234"
            }
          ]
        },
      "requestTime":"2013-06-08T11:23:37Z"
    },
    "1014":
    {
      "attributes":
        {
          "sor":"HRMS",
          "sorId":"914890374",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"089010023"
            },
            {
              "type":"network",
              "identifier":"p5478"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Richard",
              "family":"Hess"
            }
          ],
          "ou":"Biology"
        },
      "requestTime":"2013-06-08T11:23:37Z"
    }
  }
}
```

Request Resolved Matches

It may also sometimes be useful to retrieve the set of resolved match requests.

```
GET /v1/matchRequests?status=resolved

200 OK
{
  "matchRequests":
  {
    "1009":
    {
      "attributes":
        {
          "sor":"SIS",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"971194843"
            }
          ]
          "names":[
            {
              "type":"official",
              "given":"Pat",
              "family":"Lee"
            }
          ],
          "telephoneNumbers":[
            {
              "type":"mobile",
              "number":"8185551234"
            }
          ]
        },
      "requestTime":"2013-06-08T11:23:37Z",
      "resolutionTime":"2013-06-10T11:23:37Z",
      "referenceId":"M523441767"
    },
    "1014":
    {
      "attributes":
        {
          "sor":"HRMS",
          "sorId":"914890374",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"089010023"
            },
            {
              "type":"network",
              "identifier":"p5478"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Richard",
              "family":"Hess"
            }
          ],
          "ou":"Biology"
        },
      "requestTime":"2013-06-08T11:23:37Z",
      "resolutionTime":"2013-06-10T11:23:37Z"'
      "referenceId":"M523441995"
    }
  }
}
```

Request Pending Match

It may be desirable for the response to include all known SOR data, if the match engine has access to it. This would facilitate the reconciliation administrator in determining how to resolve a pending match. Such functionality is optional, and could also be provided by having the Identity Console query the Identity Registry.

The `explanation` field returned as part of a Potential Match Found response may also be returned here.

```
GET /v1/matchRequests/1009

300 Multiple Choices
{
  "candidates":
  [
    {
      "confidence":"85",
      "referenceId":"M219488003",
      "attributes":
      [
        {
          "sor":"HRMS",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"089010023"
            },
            {
              "type":"network",
              "identifier":"pl292"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Patricia",
              "family":"Lee"
            }
          ],
          "ou":"Biomedical Informatics"
        }
      ]
    },
    {
      "confidence":"71",
      "referenceId":"M523441767",
      "attributes":
      [
        {
          "sor":"guest",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"pl388"
            },
            {
              "type":"network",
              "identifier":"pl388"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Patrick",
              "family":"Lee"
            }
          ],
          "telephoneNumbers":[
            {
              "type":"mobile",
              "number":"8185551234"
            }
          ]
        }
      ]
    },
    {
      "referenceId":"new",
      "attributes":
      [
        {
          "sor":"SIS",
          "sorId":"971194843",
          "identifiers":[
            {
              "type":"sor",
              "identifier":"971194843"
            }
          ],
          "names":[
            {
              "type":"official",
              "given":"Pat",
              "family":"Lee"
            }
          ],
          "telephoneNumbers":[
            {
              "type":"mobile",
              "number":"8185551234"
            }
          ]
        }
      ]
    }
  ],
  "requestTime":"2013-06-08T11:23:37Z"
}
```

A match request that has already been resolved might not return the available candidates, but instead an indication that the match was successfully resolved.

```
GET /v1/matchRequests/1014

200 OK
{
  "requestTime":"2013-06-08T11:23:37Z",
  "resolutionTime":"2013-06-10T11:23:37Z"'
  "referenceId":"M523441995"
}
```

Reference Identifiers

It may sometimes be necessary to manually relink records, perhaps due to bad data presented at the initial match request. Directly manipulating reference identifier resources can be used to accomplish this.

The Identity Match engine may restrict access to these requests to specific administrators.

Obtain SOR Records

To obtain the set of SOR records attached to a reference identifier, submit the reference identifier in the following `GET` request.

```
GET /v1/matchRequests?referenceId=M523441767
```

The response is in the same format as for Request Resolved Matches.

If the Identity Match engine maintains additional identifiers, these may be returned as well.

```
GET /v1/matchRequests?referenceId=M523441767

200 OK
{
  "matchRequests":{
    ...
  },
  "identifiers":[
    {
      "type":"network",
      "identifier":"pl388"
    },
    {
      "type":"enterprise",
      "identifier":"905003148"
    }
  ]
}
```

In a coordinated implementation, current "golden" attributes may also be returned:

```
GET /v1/matchRequests?referenceId=M523441767

200 OK
{
  "matchRequests":{
    ...
  },
  "golden":{
    "names":[
      {
        "type":"official",
        "given":"Patricia",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
  }
}
```

This operation may also return `403 Unauthorized` or `404 Not Found`.

Join Reference Identifiers

Under certain circumstances, the Identity Match engine may not have enough information to link a new record with an existing reference identifier, resulting in a person having two (or more) reference identifiers. The reference identifiers can later be joined together using this request. The active reference identifier is provided in the resource URL and the deprecated reference identifiers in the request body.

```
PUT /v1/referenceIds/M523441767
{
  "referenceIds":[
    "M787800232",
    "M350100023"
  ]
}
```

If supported by the engine, the identifiers to maintain as primary may also be specified:

```
PUT /v1/referenceIds/M523441767
{
  "referenceIds":[
    "M787800232",
    "M350100023"
  ],
  "identifiers":[
    {
      "type":"network",
      "identifier":"pl388"
    },
    {
      "type":"enterprise",
      "identifier":"905003148"
    }
  ]
}
```

In a coordinated implementation, if supported by the engine the attributes to keep in the golden record may also be specified:

```
PUT /v1/referenceIds/M523441767
{
  "referenceIds":[
    "M787800232",
    "M350100023"
  ],
  "identifiers":[
    {
      "type":"network",
      "identifier":"pl388"
    },
    {
      "type":"enterprise",
      "identifier":"905003148"
    }
  ],
  "golden":{
    "names":[
      {
        "type":"official",
        "given":"Patricia",
        "family":"Lee"
      }
    ],
    "dateOfBirth":"1983-03-18",
  }
}
```

This operation may also return `403 Unauthorized`, `404 Not Found`, `409 Conflict` (if the engine refuses to process the requested join due to data conflicts), or `501 Not Implemented` (if identifiers or golden record attributes are provided but cannot be handled).

Reassign Reference Identifier

If the Match Engine incorrectly assigned a reference identifier to a record, it is possible for an administrator to specify a replacement reference identifier for that record. This is very similar to a Forced Reconciliation Request, except that `sorAttributes` are not specified.

```
PUT /v1/people/sis/971194843
{
  "referenceId":"M523441767"
}
```

The reference identifier can be set to `new` to indicate that a new identifier should be assigned.

This operation may also return `403 Unauthorized`, `404 Not Found`, or `409 Conflict` (if the engine refuses to process the requested update due to data conflicts).