

Patient-Import 3.x

Setup Guide for Transplant Center Partners
(Rev: February 2026)

Table of Contents

PATIENT-IMPORT OVERVIEW	4
ARCHITECTURE	4
WORK EFFORT	4
SECURITY	5
DELIVERY APPROACH	5
OVERVIEW DIAGRAM	6
PATIENT IMPORT FLOW DIAGRAM	6
SECURITY CALL	7
REQUEST PARAMETERS	7
EXAMPLES	7
PATIENT-IMPORT SERVICE CALLS	9
API VERSIONS	9
Version 2.0	9
Version 3.0	9
THE “CREATE” ENDPOINT	10
Form 117	10
HTTP Request Method	10
HTTP Headers	10
HTTP Response	10
Message Body/Patient Data	11
Required Fields	12
THE “UPDATE” ENDPOINT	16
Form 117	16
Identifying a Record	16
Removing Data from a Record	16
HTTP Request Method	16
HTTP Headers	16
HTTP Response	17
Message Body/Patient Data	17
Required Fields	18
THE “HEALTH” ENDPOINT	20
How to Use	20
HTTP Request Method	20
HTTP Headers	20
HTTP Response	20
THE “GET PATIENT STATUS” ENDPOINT	21
HTTP Request Method	21
HTTP Headers	21
HTTP Response	21
THROTTLING	21
UNKNOWN ICD-10 / DISEASE / DIAGNOSIS CODE	22
Create Call	22
Update Call	22
PATIENT-IMPORT 3.0 SPECIFICATION	23
FHIR Bundle Structure	23
Patient Info Fields	24
Lab Reference ID Field	25
Disease/Diagnosis Fields	25
Patient Weight Field	26
Blood Type (ABO) Field	27
RhD Field	29

<i>CMV Field</i>	29
<i>HLA-A Field</i>	30
<i>HLA-DPB1 Field</i>	31
LOINC CODES (VALUE SETS)	34
<i>References</i>	34
<i>FHIR HLA Format</i>	34
<i>LOINC Mapping Table</i>	35
FHIR COMPLIANCE	38
<i>Java-based FHIR Validator</i>	38
ERROR HANDLING	40
HELPFUL LINKS	43
ADDENDUM	44
SAMPLE: MINIMUM REQUIRED FOR A SUCCESSFUL “IN-PROGRESS” CREATE	44

Welcome!

Patient-Import Overview

Patient-Import is an application that allows automated patient record creation and updates and sits in front of NMDP's **MatchSource**® application. By providing a public application programming interface (API), transplant centers and partners can push data from their own healthcare systems (EPIC, CERNER, etc.) to **MatchSource**® rather than requiring users to manually copy patient fields from one system to another. This is much more efficient and decreases the chance of transposition errors.

Architecture

The public Patient-Import APIs are based on industry standards. The services expect a RESTful call over HTTPS (SSL/TLS) using the JSON data format. The message payload is based on the HL7® FHIR® healthcare standard (<https://fhir.org/>) for modeling patient data. Patient-Import has a FHIR Implementation Guide (IG) that can be referenced here: <https://fhir.nmdp.org/ig/matchsync/index.html>

A successful call to the synchronous “create” endpoint results in a response containing the patient record “RID” (*recipient identifier*, aka “NMDP ID”) that was generated. The RID can be stored in the caller’s system and is one way **MatchSource**® users can lookup patients in the UI.

A successful call to the synchronous “update” endpoint results in one or more fields being automatically modified for the patient record in NMDP systems.

Work Effort

Each Transplant Center (TC) or partner utilizing Patient-Import for patient creation/updates is free to incorporate the functions in a manner that benefits their own patient workflow. The process of mapping patient data and integrating with business process triggers is specific to each Transplant Center. For instance, one partner has built an automated system trigger to push patient records to NMDP upon receiving a BMT referral. Another has built a push-button mechanism into their healthcare system to import a patient record to **MatchSource**® manually. Because of the differences in data storage, EHR system used, and desired trigger to send data, the overall effort required to integrate with Patient-Import is unique for each TC. NMDP can provide details on past partnerships and help with scope and estimation for your organization.

Security

Patient-Import service calls are secured using JWT bearer tokens in the message header. The short-life token is obtained by the caller through a request to NMDP's identity provider. Under the oversight of the U.S. Federal Government, Department of Health and Human Services, NMDP products must comply with federal FISMA system requirements, including NIST 800-53 technical security specifications, and the technical specifications of the corresponding Authentication Assurance Level (see NIST 800-63).

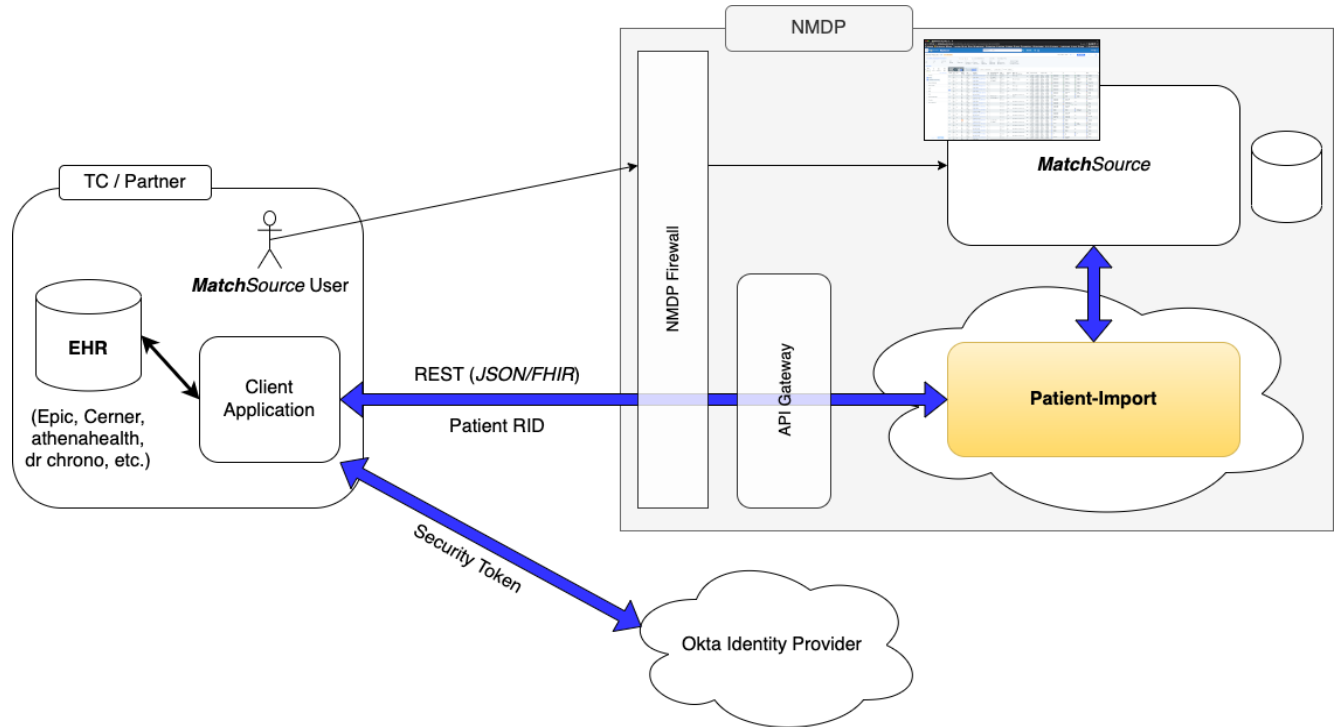
Details around Patient-Import security are discussed in the next section of this document and the NMDP security team can answer questions and address concerns as needed.

Delivery Approach

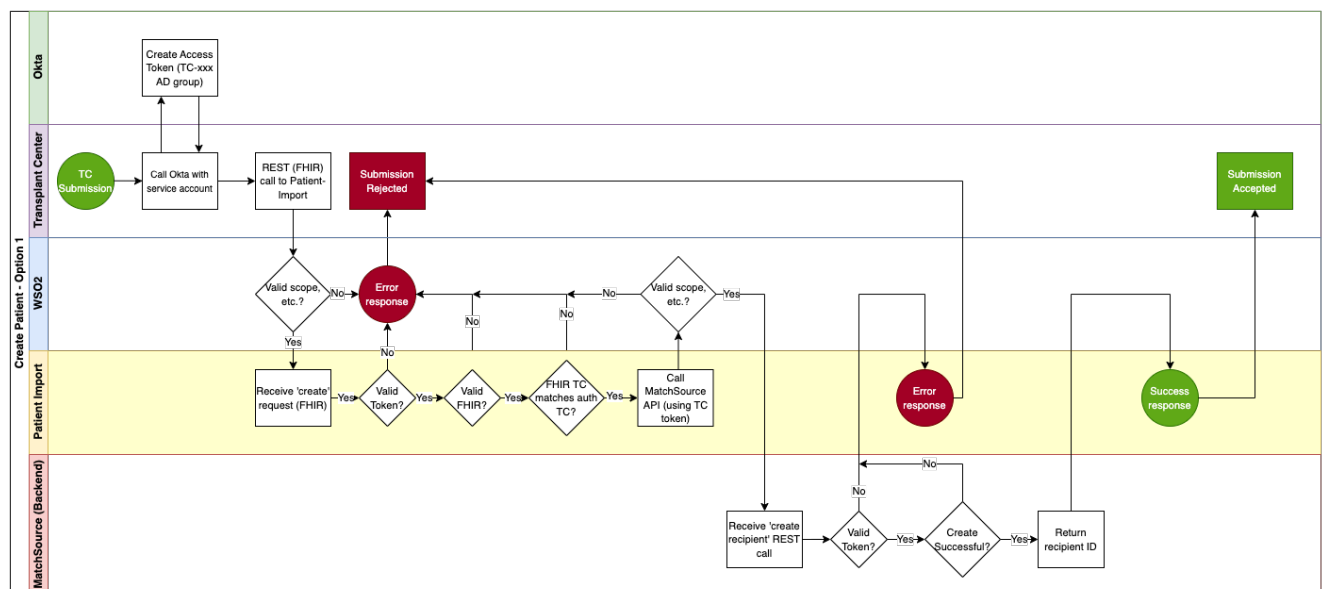
Patient-Import is exposed to our partners in a “lower” pre-production environment as well as in production. Partners may test the automation process, connections, and data integration in the lower environment. When ready, the move to production is trivial.

The NMDP delivery team works closely with partners to address issues and help troubleshoot.

Overview Diagram



Patient Import Flow Diagram



Security Call

The authentication and authorization processes protecting Patient-Import require partners to obtain a security token which is then included with the service call.

NMDP uses Okta as an identity provider; a call to Okta returns a bearer token (JWT). *Each transplant-center/partner* is issued a unique service account and password. Generated security tokens are good for **30 minutes** before expiring. It is recommended that each call to Patient-Import generate a new security token to avoid needing to manage tokens or accidentally using an expired token.

Request Parameters

There are five parameters required to obtain a bearer token:

- | | |
|----------------------------------|--|
| 1. The service account | [l1_xxx_api_user] (lower env) |
| 2. The service account password | [<account-password>] |
| 3. The application client ID | [00a1c1bzsjjzpYTcM0h8] (lower env) |
| 4. The application client Secret | [<client-secret>] |
| 5. The scope(s) needed | [int_matchsync_patient_import] (lower env) |

Note: For production, the service account name, password, and scope details will need to be updated to production values. The NMDP support team will provide these details when you are ready.

The Okta/identity provider URL for lower environment tokens is:

<https://nmdp.oktapreview.com/oauth2/ausaexcazhLhxKnJs0h7/v1/token>

The Okta/identity provider URL for production tokens is:

<https://nmdp.okta.com/oauth2/aus3ck6q30qmOdpMb1t7/v1/token>

Examples

The call to Okta can be tested with tools like Postman or command-line tools like **cUrl** as shown below:

```
$ curl --location --request POST '<okta-endpoint>' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--header 'Authorization: Basic <<base64 encoding of
<<clientId>><<clientSecret>>>>' \
--data-urlencode 'grant_type=password' \
--data-urlencode 'username=<<ServiceAccountUserID>>' \
--data-urlencode 'scope=<<Scope>>' \
--data-urlencode 'password=<<ServiceAccountPassword>>'
```

Here is what a lower environment call using one of our partner's service accounts looks like:

```
$ curl --location --request POST
'https://nmdp.oktapreview.com/oauth2/ausaexcazhLhxKnJs0h7/v1/token' \
```

[illegible]

When the JWT token is decoded, the contents of the payload are similar to this:

```
{
  "ver": 1,
  "jti":
    "AT.1WafRJEQqoZt6pBQdV_GnfgD0LbvKkfLQLAxs_yngd4",
  "iss":
    "https://nmdp.oktapreview.com/oauth2/ausaexcazhLhxKnJs0h7",
  "aud": "https://dev.authserver.nmdp.org",
  "iat": 1709768766,
  "exp": 1709855166,
  "cid": "00a1twwb1kIlt1l0h8",
  "uid": "000u1fh6aaleEPDR1J0h8",
  "scp": [
    "int_matchsync_patient_import"
  ],
  "auth_time": 1709768766,
  "sub": "l1_msk_api_user@nmdp.org",
  "matchsync_patient_import_authz": [
    "l1_role_matchsync_pi_tc_346_user",
    "l1_role_matchsync_pi_tc_347_user"
  ],
  "last_name": "_msk_api_user",
  "first_name": "l1",
  "email": "l1_msk_api_user@nmdp.org"
}
```

Patient-Import Service Calls

Once you have a valid security token from the previous section, you are authenticated and authorized to send patient “create” and “update”, and “get patient status” requests from your transplant center.

The call to Patient-Import is a standard RESTful call over HTTPS. The payload content-type is JSON (*application/json*) and an “Authorization” security header for your Bearer token must be present. The format of the JSON payload is modeled on the HL7® FHIR® healthcare standard.

The *patient-create* service call uses the HTTP method, “**POST**”. The *patient-update* service call uses the HTTP method, “**PUT**”. The *patient-status* service call uses the HTTP method, “**GET**”.

API Versions

Patient-Import hosts multiple versions of each service endpoint. Partners can choose which version best meets their needs and can decide when it makes sense to upgrade to a newer version. Incremental changes to an established version shall be non-breaking and backward compatible while new major releases (1.0, 2.0, 3.0) represent new structures or features that may require some work to upgrade.

As of April 2024, Patient-Import supports two major releases, 2.0 and 3.0. *It is highly recommended* to always target the latest API version as older versions won’t have the latest features and may be deprecated as newer versions are released.

Hosted specifications (lower environment):

Version 2.0

- Swagger: <https://int-api.nmdp.org/int-matchsource-patient-import/2.0/swagger-ui/index.html>
- API spec: <https://int-api.nmdp.org/int-matchsource-patient-import/2.0/api-docs>

Version 3.0

- **Swagger:** <https://int-api.nmdp.org/int-matchsync-patient-import/3.0/swagger-ui/index.html>
- **API spec:** <https://int-api.nmdp.org/int-matchsync-patient-import/3.0/api-docs>
- **Sample Payload:** <https://int-api.nmdp.org/int-matchsync-patient-import/3.0/example/patient-payload-example.json>
- **FHIR IG:** <https://fhir.nmdp.org/ig/matchsync/index.html>

The “Create” Endpoint

The *create* endpoint is called to automatically generate an NMDP patient record in **MatchSource**®. A successful call will create a patient record with required fields and optional demographic information, disease information, and HLA typing (yes, you can submit your patient HLA this way!).

New **MatchSource**® patient records are classified as “in-progress” if not all information is included up front. If all information, including HLA, is included in the initial submission, the record is upgraded to “preliminary” which will automatically kick off a donor search and provide search results in **MatchSource**®.

Form 117

If you submit HLA as part of this *create* call, it can satisfy the Form 117 high resolution reporting requirement. If you choose not to submit HLA to NMDP using this service, you can still depend on the existing lab/HLA intake process that assigns HLA typing data later. Note that the patient record must contain high resolution typing at HLA-A, HLA-B, HLA-C, HLA-DRB1, HLA-DPB1 and not contain serology results or any “:XX” allele codes to meet Form 117 requirements. New fields for typing date and a flag/code to certify the results as meeting Form 117 requirements have been added to the specification snippet below.

HTTP Request Method

URL (<i>production</i>)	https://api.nmdp.org/p-matchsync-patient-import/3.0/patient
URL (<i>lower/test</i>)	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/patient
Method	POST

HTTP Headers

Content-Type	application/json
Authorization	Bearer <security-token>

HTTP Response

Success	HTTP 200/OK	application/json	{ "id": "1234569", "errorMessage": null }
Success with Field(s) exceeding max	HTTP 200/OK		{ "id": "1234569", "errorMessage": "Field(s) exceeds maximum length, field(s) set to null." }
Success with Unknown ICD-10 code	HTTP 200/OK		{ "id": "1234569", "errorMessage": "An unknown diagnosis/disease code was encountered, NMDP will update our systems with the new Diagnosis value." }

Message Body/Patient Data

The message body is a JSON-formatted payload following the HL7/FHIR specification for modeling patient data, data fields, and value sets. For more information, see <https://fhir.org/> as well as some NMDP standards for FHIR as part of our implementation guide (IG): <https://fhir.nmdp.org/ig/matchsync/>

The payload includes references to your NMDP-assigned transplant center identifier. All references to *transplant center* or *partner* in the payload must match the transplant center used to authenticate. For instance, if you authenticate using a service account for transplant center “123”, references throughout the FHIR payload must also use “TC 123”, otherwise a security error will be returned. This helps to prevent one partner from inadvertently submitting data for a different transplant center.

Required Fields

The Patient-Import APIs require the minimum fields necessary to create a patient record in **MatchSource**®. Note that this is *not* necessarily enough information to execute a search in MatchSource or return search results (additional optional fields, including HLA, are needed for a preliminary search).

Patient-Import API supports all fields necessary to perform a search, including HLA, however transplant centers can still take advantage of existing NMDP processes for receiving HLA typing data from labs separately. In this case, the Patient-Import API can be used to submit all required and optional demographic fields, but then rely on HLA to be received later from the labs. Once HLA arrives, an in-progress patient in MatchSource will automatically have HLA typing assigned and will execute a preliminary search.

Required for In-Progress

This is the bare minimum needed to successfully generate an NMDP patient record. It is not enough information to run a donor search.

Element	Notes	Field Type (in-progress)
TC ID	Provided by authentication of the API	Required
Patient first name	Text string, 40 char	Required
Patient Last name	Text string, 40 char	Required
DOB	YYYY-MM-DD	Required
Local ID**	Text string (TC EHR identifier), 50 char	Recommended
Sex	Enumerated list	Required
Race	Enumerated list	Required (<i>defaulted</i>)
Ethnicity	Enumerated list	Required (<i>defaulted</i>)

****** While not strictly required, this is how most transplant centers map a patient ID in their EHR system to an NMDP patient record, so it is highly recommended that this value is always populated

See the first sample payload shown in the **Addendum** at the end of this document for details of what data was sent for this example. It only includes patient name, birthdate, local-id, sex, and transplant center identifier.

This successful response includes the NMDP ID (RID) for the created record.

Body
200 OK 3.42 s 1.12 KB

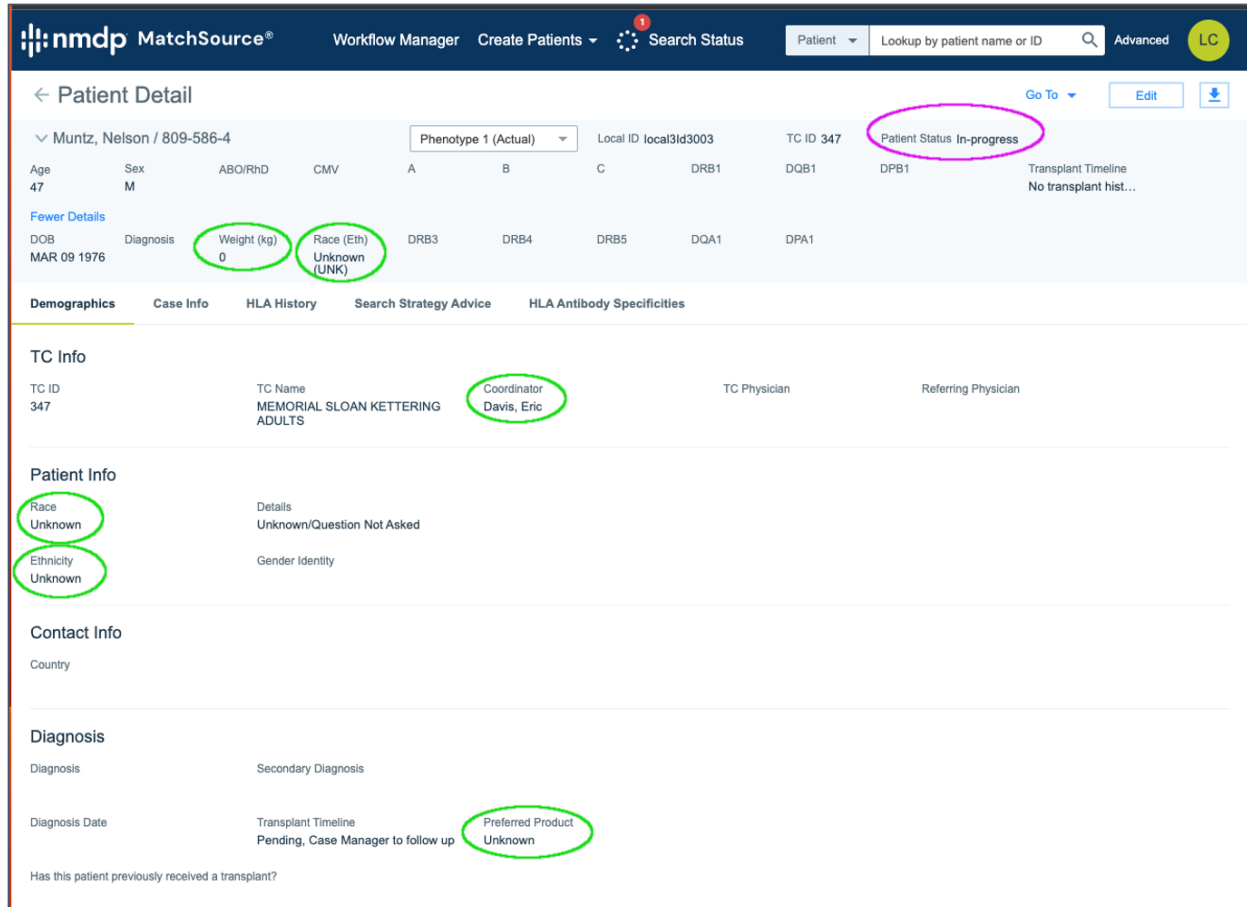
Pretty
Raw
Preview
Visualize

```

{"id":"8095864"}

```

When viewed in **MatchSource**[®], this minimum-data *create* call produces the following record:



nmdp MatchSource[®] Workflow Manager Create Patients Search Status Patient Lookup by patient name or ID Advanced LC

← Patient Detail Go To Edit

▼ Muntz, Nelson / 809-586-4 Phenotype 1 (Actual) Local ID local3ld3003 TC ID 347 Patient Status In-progress

Age	Sex	ABO/RHD	CMV	A	B	C	DRB1	DQB1	DPB1	Transplant Timeline
47	M									No transplant hist...

[Fewer Details](#)

DOB	Diagnosis	Weight (kg)	Race (Eth)	DRB3	DRB4	DRB5	DQA1	DPA1
MAR 09 1976		0	Unknown (UNK)					

Demographics Case Info HLA History Search Strategy Advice HLA Antibody Specificities

TC Info

TC ID	TC Name	Coordinator	TC Physician	Referring Physician
347	MEMORIAL SLOAN KETTERING ADULTS	Davis, Eric		

Patient Info

Race	Details
Unknown	Unknown/Question Not Asked

Ethnicity	Gender Identity
Unknown	

Contact Info

Country

Diagnosis

Diagnosis	Secondary Diagnosis

Diagnosis Date	Transplant Timeline	Preferred Product
	Pending, Case Manager to follow up	Unknown

Has this patient previously received a transplant?

Be aware that if not included, some fields are defaulted. The fields shown in **green** above are some examples of defaulted fields, which include weight, race, ethnicity, TC coordinator, etc.

Also note (**magenta**), the status of this patient record is “In-Progress” which represents a partial record (draft status) and does not include enough information to be able to kick off a donor search.

Required for Preliminary Search

These are the fields needed to both create a new patient record and generate a donor search in **MatchSource**®. If successful, records created with the fields below will show status “PRLM”.

Element	Notes	Field Type (PRLM)
TC ID	Provided by authentication of the API	Required
Secondary Code	TC org abbreviation	Optional
TC coordinator	From enumerated list (MatchSource ®)	Required (<i>defaulted</i>)
TC Physician: FN LN	Text string, 24 char	Optional
Referring Physician: FN LN	Text string, 24 char	Optional
Patient first name	Text string, 40 char	Required
Patient Last name	Text string, 40 char	Required
DOB	YYYY-MM-DD	Required
Local ID	Text string (TC EHR identifier), 50 char	Required
ITL/RefID for lab	ITL - Lab HLA reference id, 50 char	Optional (required for traditional HLA intake)
Sex	Enumerated list	Required
Weight	Positive integer (kg)	Optional (<i>defaulted</i>)
Race	Enumerated list	Required (<i>defaulted</i>)
Ethnicity	Enumerated list	Required (<i>defaulted</i>)
ABO (blood type)	Enumerated list	Optional
RH factor	Enumerated list	Optional
CMV	Enumerated list	Optional
Country	Enumerated list	Required
Street Address	Only if Country is US	Required (if US)
City		Required (if US)
State	Enumerated list	Required (if US)
Zip	Not validated	Required (if US)
Phone	10 char	Required (if US, <i>defaulted</i>)
Email	50 char max	Optional
Preferred Language	Enumerated list	Required (if US, <i>defaulted</i>)
Diagnosis	Enumerated list	Required
Diagnosis Date		Required
Secondary Diagnosis		Conditionally Required
Disease Stage	Enumerated list	Conditionally Required, <i>defaulted</i>
Number of Remissions		Optional (<i>defaulted</i>)
Transplant Timeline	Enumerated list	Required (<i>defaulted</i>)

Preferred Product	Enumerated list	Required (<i>defaulted</i>)
HLA-A [†]		Required to Search
HLA-B [†]		Required to Search
HLA-C		Optional to Search
HLA-DRB1 [†]		Required to Search
HLA-DQB1		Optional to Search
HLA-DPB1		Optional to Search
HLA-DRB3		Optional to Search
HLA-DRB4		Optional to Search
HLA-DRB5		Optional to Search
HLA-DQA1		Optional to Search
HLA-DPA1		Optional to Search

[†] *Minimum required HLA to perform a donor search*

Setup data for the enumerated fields (e.g. diagnosis) will be supplied by the Patient-Import delivery team. This is one of the first topics covered when meeting with new users to go over the value sets expected in **MatchSource**[®], how they are modeled in FHIR (and the value code system where applicable), and what mappings might be needed.

The “Update” Endpoint

The *update* endpoint is called to modify an existing NMDP patient record in **MatchSource**®. A successful call will cause one or more fields to be updated including demographic information, disease information, and HLA typing. Any patient record can be updated using this API endpoint, not just patient records created through Patient-Import.

MatchSource® patient records are classified as “in-progress” if not enough data is included to be able to run a donor search. If all information, including HLA, is present following an *update*, the record is upgraded to “preliminary” which will automatically kick off a donor search and display search results in **MatchSource**®.

Form 117

If you submit HLA as part of an *update* call, it can satisfy the Form 117 high resolution reporting requirement. By sending a typing date and a value certifying your HLA meets Form 117 high resolution requirements, you need not submit a Form 117 separately. Note that the patient record must contain high resolution typing at HLA-A, HLA-B, HLA-C, HLA-DRB1, HLA-DPB1 and not contain serology results or any “:XX” allele codes to be considered for Form 117 compliance. New fields for typing date and a flag/code to certify the results as meeting Form 117 requirements have been added to the specification snippet below.

Identifying a Record

When an “update” is performed, the target record is identified by three pieces of data:

- Transplant **center code** (3-digit number assigned by NMDP) – to be included in the payload and as part of the bearer token used to authenticate
- Patient **RID** (NMDP recipient ID) – the 7-digit RID returned by the “create” endpoint (or as displayed in **MatchSource**®)
- Patient **date-of-birth** – sent as an extra check to make sure the RID submitted is the correct target patient

Removing Data from a Record

Existing data in optional fields can be removed by passing *null*. Exceptions to this are ABO, RhD, CMV and phone number which can be replaced but cannot be removed once set. See the “Nullable” column under “*Required Fields*” below.

HTTP Request Method

URL (<i>production</i>)	https://api.nmdp.org/p-matchsync-patient-import/3.0/ patient/{RID}
URL (<i>lower/test</i>)	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/ patient/{RID}
Method	PUT

HTTP Headers

Content-Type	application/json
Authorization	Bearer <security-token>

HTTP Response

Success	HTTP 200/OK	application/json	{ "id": "1234569", "errorMessage": null }
Success with Field(s) exceeding max	HTTP 200/OK	application/json	{ "id": "1234569", "errorMessage": "Field(s) exceeds maximum length, field(s) set to null." }
Success with Unknown ICD-10 code when Patient is IN-PROGRESS	HTTP 200/OK	application/json	{ "id": "1234569", "errorMessage": "An unknown diagnosis/disease code was encountered, NMDP will update our systems with the new Diagnosis value." }
Fail with Unknown ICD-10 code when Patient is PRLM/FRML/ CLOSED in Matchsource	HTTP 400/ BAD REQUEST	application/json	{ "code": 400, "message": "Patient data does not meet requirements to register in MatchSource, please provide correct data for [Condition code.coding.code(UnknownIcd10Code)]" }

Message Body/Patient Data

The “update” feature uses the same FHIR structure and field definitions as the “create” endpoint.

The message body is a JSON-formatted payload following the HL7/FHIR specification for modeling patient data, data fields, and value sets. For more information, see <https://fhir.org/> as well as some NMDP standards for FHIR as part of our implementation guide (IG): <https://fhir.nmdp.org/ig/matchsync/>

The payload includes references to your NMDP-assigned transplant center identifier. All references to *transplant center* or *partner* in the payload must match the transplant center used to authenticate. For instance, if you authenticate using a service account for transplant center 123, references throughout the FHIR payload must also use “TC 123”, otherwise a security error will be returned. This helps to prevent one partner from inadvertently submitting data for a different transplant center.

Required Fields

Following an “*update*” call, the resulting patient record is validated against the list of required fields and enumerated field values. If the list of required fields is not met, the update operation fails.

For a list of required fields enforced when updating an “in-progress” record, see the earlier [Required for In-Progress](#) section of this document under the “*create*” API.

For records being updated as “preliminary”, the below fields are considered required.

Element	Notes	Nullable?	Field Type (PRLM)
TC ID	Provided by authentication of the API	N	Required
Secondary Code	TC org abbreviation	Y	Optional
TC Coordinator	From enumerated list (MatchSource ®)	(Default: first)	Required (defaulted)
TC Physician: FN LN	Text string, 24 char	Y	Optional
Referring Physician: FN LN	Text string, 24 char	Y	Optional
Patient first name	Text string, 40 char	N	Required
Patient Last name	Text string, 40 char	N	Required
DOB	YYYY-MM-DD	N	Required
Local ID	Text string (TC EHR identifier), 50 char		Required
ITL/RefID for lab	ITL - Lab HLA reference id, 50 char	Y	Optional (required for traditional HLA intake)
Sex	Enumerated list	N	Required
Weight	Positive integer (kg)	Y (Default: 0)	Optional (defaulted)
Race	Enumerated list	(Default: “Unknown”)	Required (defaulted)
Ethnicity	Enumerated list	(Default: “Unknown”)	Required (defaulted)
ABO (blood type)	Enumerated list	N	Optional
RH factor	Enumerated list	N	Optional
CMV	Enumerated list	N	Optional
Country	Enumerated list	N	Required
Street Address	Only if Country is US	N	Required (if US)
City		N	Required (if US)
State	Enumerated list	N	Required (if US)
Zip	Not validated	N	Required (if US)
Phone	10 char	N	Required (if US)
Email	50 char max	Y	Optional
Preferred Language	Enumerated list	(Default: “English”)	Required (defaulted)
Diagnosis	Enumerated list	N	Required (ICD-10 mapping)

Diagnosis Date		N	Required
Secondary Diagnosis			Conditionally Required
Disease Stage	Enumerated list		
Number of Remissions		(Default: "99")	Optional (defaulted)
Transplant Timeline	Enumerated list	(Default: "Pending")	Required (defaulted)
Preferred Product		(Default: "Unknown")	Required (defaulted)
HLA-A [†]		N	Required to Search
HLA-B [†]		N	Required to Search
HLA-C			Optional to Search
HLA-DRB1 [†]		N	Required to Search
HLA-DQB1			Optional to Search
HLA-DPB1			Optional to Search
HLA-DRB3			Optional to Search
HLA-DRB4			Optional to Search
HLA-DRB5			Optional to Search
HLA-DQA1			Optional to Search
HLA-DPA1			Optional to Search

[†] *Minimum required to perform a donor search*

The “Health” Endpoint

The *health* endpoint is used to determine if the Patient-Import application is available to create or modify patient records. It can be used as a “heartbeat” endpoint by transplant centers and partners. The Patient-Import application has a target of 99.9% (“three nines”) availability and is expected to always be available during business hours, 7 am to 8 pm US central time, Monday through Friday.

NMDP has maintenance windows, security patching, and other activities that will cause Patient-Import or its dependent systems to be down from time to time, for example after 9 PM central time on certain Fridays. This *health* endpoint may optionally be used to verify the status of Patient-Import before sending a patient *create* or *update* request.

An HTTP 200/OK response means Patient-Import and **MatchSource**® are available and healthy. Any other response should be interpreted as Patient-Import being unable to process new requests. In the event of a failure, try again in a few minutes.

This call does not require a security token or header.

How to Use

Two ways this endpoint can be used are *on-demand* and as part of a health check *polling loop*. The on-demand approach would be suitable for a system where patient records are manually submitted to NMDP and could determine whether a trigger or submission button is enabled or disabled. A polling loop would work well with a fully automated system where the client periodically checks the health of Patient-Import (e.g. every 60 seconds) and queues submissions until a success response is returned.

HTTP Request Method

URL (<i>production</i>)	https://api.nmdp.org/p-matchsync-patient-import/3.0/health
URL (<i>lower/test</i>)	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/health
Method	GET

HTTP Headers

N/A	(No security token required)
-----	------------------------------

HTTP Response

Success / Ready	HTTP 200/OK	application/json	{ "status": "UP" }
Patient-Import systems unable to process requests	HTTP 500/Internal Server Error	application/json	{ "status": "DOWN" }
Patient-Import down or unavailable	HTTP 404/Not Found HTTP 401/Unauthorized		

The “GET Patient Status” Endpoint

The “*patient status*” endpoint provides the ability to retrieve the NMDP status of a patient for the NMDP ID (RID) passed in. A successful call returns the patient NMDP ID, one of four NMDP patient statuses, and your transplant center identifier. If Patient Import is not able to find a patient for the NMDP ID, an error is returned.

NMDP patient status will be one of “**In-progress**” (draft), “**PRLM**” (preliminary), “**FRML**” (formal), or “**Closed**”.

HTTP Request Method

URL (<i>production</i>)	https://api.nmdp.org/p-matchsync-patient-import/3.0/ patient/{id}/status
URL (<i>lower/test</i>)	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/ patient/{id}/status
Method	GET

HTTP Headers

Content-Type	application/json
Authorization	Bearer <security-token>

HTTP Parameters

id	Path variable	7-digit patient ID assigned by NMDP
----	---------------	-------------------------------------

HTTP Response

Success	HTTP 200/OK	application/json	{ "tcId": "123", "id": 1234569, "patientStatus": "<patient status>" }
---------	-------------	------------------	--

Throttling

For added security and to guard against accidental large volumes of submissions hitting the Patient-Import create/update endpoints, each transplant center and partner is limited to **10 calls** within a **5-minute** period.

Note that the “/health” endpoint is not included in this limit.

If you believe your institution will exceed this limit, please contact the support team.

Unknown ICD-10 / Disease / Diagnosis Code

Due to the large number of ICD-10 codes, the MatchSync Reference Data does not hold all possible codes, but instead only codes that have already been requested by Users. In the event a User sends a payload containing an ICD-10 code that is not present in the MatchSync Reference Data, an Email will be sent to an internal MatchSync team to identify and catalog the new ICD-10 code. The same team will then edit the Patient in MatchSource with the correct NMDP Diagnosis Code. New code(s) identified will be added to the MatchSync Reference Data after a few days.

Create Call

If an error occurs when attempting to CREATE a new patient, the Patient will be created in MatchSource without a diagnosis code and will be given a status of “IN-PROGRESS”. An internal MatchSync Team member will edit the Diagnosis in MatchSource once the new code is identified. If any other validation failures occur, the patient will not be created, and a HTTP 400 Response will be returned instead.

Update Call

If an error occurs when attempting to UPDATE a Patient in MatchSource, the result will depend on the current Status of the Patient.

If the Patient has a Status of IN-PROGRESS, the update will return an HTTP 200 response with an error message. The patient Diagnosis will be blank in MatchSource but will be filled in by a MatchSync team member.

If the Patient has a Status of PRLM, FRML, or CLOSED, an HTTP 400 response will be returned, and no updates will be made to the Patient in MatchSource.

Patient-Import 3.0 Specification

A sample payload is publicly available and hosted by NMDP at: <https://int-api.nmdp.org/int-matchsync-patient-import/3.0/example/patient-payload-example.json>

The FHIR implementation guide can be found here:
<https://fhir.nmdp.org/ig/matchsync/artifacts.html>

The following is a detailed look at the FHIR payload structure and a few specific examples. Patient-Import only expects **one** patient record per submission in the FHIR bundle.

FHIR Bundle Structure

```
{
  "resourceType": "Bundle",
  "meta": {
    "profile": ["http://fhir.nmdp.org/ig/matchsource/StructureDefinition/msbundle"],
    "security": [
      {
        "system": "http://terminology.nmdp.org/codesystem/transplant-center",
        "code": "tc_123"
      }
    ]
  },
  "type": "collection",
  "timestamp": "2020-11-24T23:50:50-05:00",
  "entry": [
    {
      "fullUrl": "urn:uuid:cc111e78-15eb-430d-b337-a4418494bedc",
      "resource": {...}
    },
    {
      "fullUrl": "urn:uuid:cef4702d-cbe1-447b-b671-588dbfb40bb5",
      "resource": {...}
    },
    {
      "fullUrl": "urn:uuid:8fd0ef20-d146-4fff-b2a8-1d607adf9802",
      "resource": {...}
    },
    {
      "fullUrl": "urn:uuid:8fd0ef20-d146-4fff-b2a8-1d607adf9802",
      "resource": {...}
    },
    {
      "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f1",
      "resource": {...}
    },
    ...
    {
      "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62fe",
      "resource": {...}
    }
  ]
}
```

The 3-digit transplant center code (in yellow) is unique to your organization.

Patient Info Fields

Patient Entry

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-mspatient.html>

```
{
  "fullUrl": "urn:uuid:cc111e78-15eb-430d-b337-a4418494bedc",
  "resource": {
    "resourceType": "Patient",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/mspatient"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "name": [
      {
        "family": "Muntz",
        "given": [
          "Nelson",
          "B"
        ]
      }
    ],
    "gender": "male",
    "birthDate": "1976-03-09",
    "telecom": [
      {
        "system": "phone",
        "value": "1-612-555-1234"
      },
      {
        "system": "email",
        "value": "abc@tc.hospital.org"
      }
    ],
    "address": [
      {
        "line": [
          "123 Main St"
        ],
        "city": "Minneapolis",
        "state": "MN",
        "postalCode": "55401",
        "country": "US"
      }
    ],
    "identifier": [
      {
        "use": "usual",
        "system": "http://terminology.nmdp.org/identifier/local-123",
        "value": "local3EpicId3003"
      }
    ],
    "communication": [
      {
        "language": {
          "coding": [
            {
              "system": "urn:ietf:bcp:47",
              "code": "ENG",
              "display": "English"
            }
          ]
        },
        "preferred": true
      }
    ]
  }
}
```

```

    },
    "managingOrganization": {
      "reference": "urn:uuid:413d5929-1fc5-43a4-ad48-e7ffe6af768c"
    }
  }
}

```

Lab Reference ID Field

LabRefId – This entry can be omitted if HLA is included in the Patient-Import *create* or *update* payloads. If not, this lab reference value is used to map HLA lab results to a patient record at a later time.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-msspecimen.html>

```

{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f1",
  "resource": {
    "resourceType": "Specimen",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/msspecimen"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "identifier": [
      {
        "use": "usual",
        "system": "http://terminology.nmdp.org/identifier/lab-123",
        "value": "labRef21Id123"
      }
    ],
    "subject": {
      "reference": "urn:uuid:cc111e78-15eb-430d-b337-a4418494bedc"
    }
  }
}

```

Disease/Diagnosis Fields

Fields to report the disease/diagnosis, secondary diagnosis, disease stage, and remissions. Disease codes are based on **ICD-10 codes** and mapped by NMDP to the correct enumerated fields in **MatchSource**®.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-msdiagnosis.html>

<https://fhir.nmdp.org/ig/matchsync/CodeSystem-nmdp-disease-cs.html>

```

{
  "fullUrl": "urn:uuid:be7873b5-9685-42a9-9815-6d048e6dcc8f",
  "resource": {
    "resourceType": "Condition",
    "id": "MSDiagnosisExample",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/msdiagnosis"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "extension": [
      {

```

```

    "url": "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/number-of-remissions",
    "valueInteger": 0
  },
  {
    "url": "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/secondary-diagnosis",
    "valueString": "Some Secondary Diagnosis"
  },
  {
    "url": "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/other-diagnosis",
    "valueString": "Not Provided"
  }
],
"code": {
  "coding": [
    {
      "system": "http://hl7.org/fhir/sid/icd-10-cm",
      "code": "C86.50",
      "display": "Non-Hodgkin Lymphoma"
    }
  ]
},
"subject": {
  "reference": "Patient/MSPatientExample"
},
"recordedDate": "2021-11-01",
"stage": [
  {
    "summary": {
      "coding": [
        {
          "system": "http://terminology.nmdp.org/codesystem/diseasestage",
          "code": "RM",
          "display": "Remission"
        }
      ]
    }
  ]
}
]
}
}
}
}
}

```

Patient Weight Field

Field to report patient weight, in kilograms.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-ms-weight-observation.html>

```

{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f0",
  "resource": {
    "resourceType": "Observation",
    "id": "MSWeightObsExample",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/ms-weight-observation"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "status": "final",
    "category": [
      {
        "coding": [
          {
            "system": "http://terminology.hl7.org/CodeSystem/observation-category",
            "code": "vital-signs"
          }
        ]
      }
    ]
  }
}

```

```

    }
  ],
  "code": {
    "coding": [
      {
        "system": "http://loinc.org",
        "code": "29463-7",
        "display": "Body weight"
      }
    ]
  },
  "subject": {
    "reference": "Patient/MSPatientExample"
  },
  "effectiveDateTime": "",
  "valueQuantity": {
    "value": 70,
    "unit": "Kg",
    "system": "",
    "code": ""
  }
}

```

Blood Type (ABO) Field

Field to report patient ABO.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-ms-abogroup-observation.html>

```

{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f0",
  "resource": {
    "resourceType": "Observation",
    "id": "MSABOObsExample",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/ms-abogroup-
observation"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "status": "final",
    "code": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "85504-9",
          "display": "Presumptive ABO group [Type] in Saliva (oral fluid) by Molecular
genetics method"
        }
      ]
    },
    "subject": {
      "reference": "Patient/MSPatientExample"
    },
    "effectiveDateTime": "",
    "valueCodeableConcept": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "LA19710-5",
          "display": "Group A"
        }
      ]
    }
  }
}

```


RhD Field

Field to report patient Rh value for blood type.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-ms-rhstatus-observation.html>

```
{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f0",
  "resource": {
    "resourceType": "Observation",
    "id": "MSRhObsExample",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/ms-rhstatus-observation"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "status": "final",
    "code": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "85505-6",
          "display": "Presumptive ABO group [Type] in Saliva (oral fluid) by Molecular genetics method"
        }
      ]
    },
    "subject": {
      "reference": "Patient/MSPatientExample"
    },
    "effectiveDateTime": "2022-05-01",
    "valueCodeableConcept": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "LA27326-0",
          "display": "Rh+"
        }
      ]
    }
  }
}
```

CMV Field

Field to report patient CMV status.

```
{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f0",
  "resource": {
    "resourceType": "Observation",
    "id": "CMV",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/cibmtr-reporting/StructureDefinition/cibmtr-obs-priority-variables"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "status": "final",
```

```

    "code": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "13949-3",
          "display": "Cytomegalovirus IgG Ab [Presence] in Serum or Plasma by
Immunoassay"
        }
      ],
    },
    "subject": {
      "reference": "Patient/MSPatientExample"
    },
    "effectiveDateTime": "2022-05-01",
    "valueCodeableConcept": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "LA6576-8",
          "display": "Positive"
        }
      ]
    }
  }
}

```

HLA-A Field

Field to report patient HLA for locus A.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-hla-a.html>

```

{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f4",
  "resource": {
    "resourceType": "Observation",
    "id": "HLA-A",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/hla-genotype"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "status": "final",
    "category": [
      {
        "coding": [
          {
            "system": "http://terminology.hl7.org/CodeSystem/observation-category",
            "code": "laboratory"
          }
        ]
      }
    ],
    "code": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "84413-4"
        },
        {
          "system": "http://loinc.org",
          "code": "13298-5",
          "display": "HLA-A [Type]"
        }
      ]
    }
  }
}

```

```

    },
    "subject": {
      "reference": "Patient/MSPatientExample"
    },
    "effectiveDateTime": "2022-05-01",
    "valueCodeableConcept": {
      "coding": [
        {
          "system": "http://hlstring.org",
          "code": "hla#3.25.0#HLA-A*01:01:01:01+HLA-A*01:02"
        }
      ]
    },
    "specimen": {
      "reference": "Specimen/MSSpecimenExample"
    },
    "component": [
      {
        "code": {
          "coding": [
            {
              "system": "http://loinc.org",
              "code": "48018-6"
            }
          ]
        },
        "valueCodeableConcept": {
          "coding": [
            {
              "system": "http://www.genenames.org/geneId",
              "code": "HGNC:4931",
              "display": "HLA-A"
            }
          ]
        }
      }
    ]
  }
}

```

HLA-DPB1 Field

Field to report patient HLA for locus DPB1.

<https://fhir.nmdp.org/ig/matchsync/StructureDefinition-hla-dpb1.html>

```

{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62fe",
  "resource": {
    "resourceType": "Observation",
    "id": "HLA-DPB1",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/hla-genotype"
      ],
      "security": [
        {
          "system": "http://terminology.nmdp.org/codesystem/transplant-center",
          "code": "tc_123"
        }
      ]
    },
    "status": "final",
    "category": [
      {
        "coding": [
          {
            "system": "http://terminology.hl7.org/CodeSystem/observation-category",
            "code": "laboratory"
          }
        ]
      }
    ]
  }
}

```

```

    },
    "code": {
      "coding": [
        {
          "system": "http://loinc.org",
          "code": "84413-4"
        },
        {
          "system": "http://loinc.org",
          "code": "59017-4",
          "display": "HLA-DPB1 [Type] by High resolution"
        }
      ]
    },
    "subject": {
      "reference": "Patient/MSPatientExample"
    },
    "effectiveDateTime": "2022-05-01",
    "valueCodeableConcept": {
      "coding": [
        {
          "system": "http://glstring.org",
          "code": "hla#3.25.0#HLA-DPB1*04:01:01G+HLA-DPB1*04:02:01G"
        }
      ]
    },
    "specimen": {
      "reference": "Specimen/MSSpecimenExample"
    },
    "component": [
      {
        "code": {
          "coding": [
            {
              "system": "http://loinc.org",
              "code": "48018-6"
            }
          ]
        },
        "valueCodeableConcept": {
          "coding": [
            {
              "system": "http://www.genenames.org/geneId",
              "code": "HGNC:4931",
              "display": "HLA-DPB1"
            }
          ]
        }
      }
    ]
  }
}

```

Form117 Fields

Field to report whether a patient is meeting form117 compliance requirements.

```

{
  "fullUrl": "urn:uuid:c3a35ccd-26aa-4a2d-a558-b382b8ea62f1",
  "resource": {
    "resourceType": "Observation",
    "id": "MSForm117ComplianceExample",
    "meta": {
      "profile": [
        "http://fhir.nmdp.org/ig/matchsource/StructureDefinition/ms-form-117-compliance-observation"
      ],
      "security": [
        {

```

```

        "system": "http://terminology.nmdp.org/codesystem/transplant-center",
        "code": "tc_123"
    }
  ],
  "status": "final",
  "code": {
    "coding": [
      {
        "system": "http://terminology.nmdp.org/codesystem/nmdp",
        "code": "FORM117COMPLIANCE",
        "display": "Form 117 Compliance"
      }
    ],
    "text": "Form 117 Compliance"
  },
  "subject": {
    "reference": "Patient/MSPatientExample"
  },
  "effectiveDateTime": "2022-05-01",
  "valueBoolean": true
}

```

LOINC Codes (value sets)

NMDP uses LOINC (loinc.org) as the reference code systems for the FHIR 'question' and 'answer' sets when reporting observations like HLA, blood type, CMV status, and more.

References

Gene Group codes for HGNC can be found

here: <https://www.genenames.org/data/genegroup/#!/group/588>

A Patient Import sample payload which includes HLA reporting: <https://int-api.nmdp.org/int-matchsync-patient-import/3.0/example/patient-payload-example.json>

LOINC codes can be viewed at <https://loinc.org/<code>>

FHIR HLA Format

The HLA **question** set in FHIR has the format:

```
"code" : {
  "coding" : [
    {
      "system" : http://loinc.org,
      "code" : "84413-4"
    },
    {
      "system" : http://loinc.org,
      "code" : "<LOINC HLA QUESTION CODE>",
      "display" : "<HUMAN READABLE TEXT>"
    }
  ]
}
```

The HLA **answer** set has the format:

```
"valueCodeableConcept" : {
  "coding" : [
    {
      "system" : http://glstring.org,
      "code" : "<GLString FOR GENOTYPE PAIR>"
    }
  ],
  "component" : [
    {
      "code" : {
        "coding" : [
          {
            "system" : http://loinc.org,
            "code" : "48018-6"
          }
        ]
      }
    }
  ]
}
```

```

    ]
  },
  "valueCodeableConcept" : {
    "coding" : [
      {
        "system" : http://www.genenames.org/geneId,
        "code" : "<HGNC GENE GROUP>",
        "display" : "<HUMAN READABLE GENE GROUP>"
      }
    ]
  }
}

```

LOINC Mapping Table

FHIR Field	NMDP-Suggested Question Code	NMDP-Suggested Answer Set
ABO	85504-9 (Presumptive ABO group)	LA19710-5 (Group A) LA19709-7 (Group B) LA19708-9 (Group O) LA28449-9 (Group AB)
RhD	85505-6 (Presumptive Rh)	LA27326-0 (Rh+) LA27325-2 (Rh-)
ABO + RhD	(let us know if this is needed)	(lab dependent)
CMV	31797-4 (Cytomegalovirus Ag)	LA6576-8 (Positive) LA6577-6 (Negative) LA9663-1 (Inconclusive) LA13538-6 (Not tested) - Untested
HLA-A	84413-4 (Genotype) 13298-5 (HLA-A)	48018-6 (Gene studied) valueCodeableConcept: { "system": http://www.genenames.org/geneId , "code": "HGNC:4931", "display": "HLA-A" }
HLA-B	84413-4 (Genotype) 13299-3 (HLA-B)	48018-6 (Gene studied) valueCodeableConcept: { "system": http://www.genenames.org/geneId , "code": "HGNC:4931", "display": "HLA-B" }
HLA-C	84413-4 (Genotype) 13302-5 (HLA-C)	48018-6 (Gene studied) valueCodeableConcept:

FHIR Field	NMDP-Suggested Question Code	NMDP-Suggested Answer Set
		<pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-C" }</pre>
HLA-DRB1	84413-4 (Genotype) 57298-2 (HLA-DRB1)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DRB1" }</pre>
HLA-DQB1	84413-4 (Genotype) 53938-7 (HLA-DQB1)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DQB1" }</pre>
HLA-DPB1	84413-4 (Genotype) 59017-4 (HLA-DPB1)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DPB1" }</pre>
HLA-DRB3	84413-4 (Genotype) 59022-4 (HLA-DRB3)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DRB3" }</pre>
HLA-DRB4	84413-4 (Genotype) 59021-6 (HLA-DRB4)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DRB4" }</pre>
HLA-DRB5	84413-4 (Genotype) 59020-8 (HLA-DRB5)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", }</pre>

FHIR Field	NMDP-Suggested Question Code	NMDP-Suggested Answer Set
		<pre>"display": "HLA-DRB5" }</pre>
HLA-DQA1	84413-4 (Genotype) 59019-0 (HLA-DQA1)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DQA1" }</pre>
HLA-DPA1	84413-4 (Genotype) 59018-2 (HLA-DPA1)	48018-6 (Gene studied) valueCodeableConcept: <pre>{ "system": http://www.genenames.org/geneId, "code": "HGNC:4931", "display": "HLA-DPA1" }</pre>

FHIR Compliance

There are many tools that can be used to validate your FHIR payload, and you can call the NMDP lower environment instance of Patient-Import as often as you like. In addition, here are instructions for installing a java-based validator. This solution takes advantage of the publicly available HL7 FHIR validator along with NMDP-specific FHIR constructs.

Java-based FHIR Validator

Prerequisite:

- GitHub account
- Access to the DASH repo (contact the NMDP team for details)

Download resources:

Download validator jar from <http://hl7.org/fhir/validator/> and save it in local directory like documents.

Clone the dash repo from <https://github.com/nmdp-bioinformatics/dash>

Copy the profiles folder from dash/dassh4/validation folder and put it in local directory like documents.

Run the JAR

Go to the location where validation jar is saved and enter below command:

```
java -jar validator_cli.jar <location of FHIR payload json to check> -  
version 4.0.1 -ig <location of the profiles folder>
```

```
Example: java -jar validator_cli.jar /Users/akhan/Documents/PayloadFor2_0.json -  
version 4.0.1 -ig ./profiles/
```

Once you run the jar you should see all warnings and errors.

```
A0507168:DevApps akhan$ java -jar validator_cli.jar /Users/akhan/Documents/PayloadFor2.0.json -version 4.0.1 -ig ./profiles/
FHIR Validation tool Version 5.6.75 (Git# a285132f5b46). Built 2022-10-28T11:57:44.269Z (47 days old)
Java: 11.0.6 from /Library/Java/JavaVirtualMachines/adaptopenjdk-11.jdk/Contents/Home on x86_64 (64bit). 4096MB available
Paths: Current = /Users/akhan/DevApps, Package Cache = /Users/akhan/.fhir/packages
Params: /Users/akhan/Documents/PayloadFor2.0.json -version 4.0.1 -ig ./profiles/
Jurisdiction: No Jurisdiction

Loading
  Load FHIR v4.0 from hl7.fhir.r4.core#4.0.1 Load hl7.terminology.r4#5.0.0 - 4174 resources (00:06.929)
  - 7394 resources (00:00.001)
  Load hl7.terminology#4.0.0 - 4164 resources (00:01.777)
  Load R5 Extensions - 123 resources (00:07.690)
  Terminology server http://tx.fhir.org - Version 2.0.14 (00:00.698)
  Load ./profiles/+ ... load IG from hl7.terminology.r4#4.0.0
+ .. load IG from hl7.fhir.us.core#4.0.0
+ .. load IG from hl7.fhir.uv.bulkdata#1.0.1
+ .. load IG from us.nlm.vsac#0.3.0
+ .. load IG from hl7.fhir.us.mcode#2.0.0
+ .. load IG from hl7.fhir.us.core#4.0.0
+ .. load IG from hl7.fhir.uv.bulkdata#1.0.1
+ .. load IG from us.nlm.vsac#0.3.0
+ .. load IG from hl7.fhir.uv.genomics-reporting#1.0.0
- 12868 resources (00:34.524)
Get set... go (00:00.007)
Validating
  Validate /Users/akhan/Documents/PayloadFor2.0.json
  Validate Bundle against http://hl7.org/fhir/StructureDefinition/Bundle.....20.....40.....60.....80.....1
  00:03.711
Done. Times: Loading: 00:51.909, validation: 00:03.711. Memory = 2Gb

*FAILURE*: 30 errors, 42 warnings, 45 notes
Error @ Bundle.entry[0].resource.ofType(Patient).extension[1].extension[0].value.ofType(Coding) (line 51, col83): The Coding provided (http://terminology.hl7.org/CodeSystem/v3-NullFlavor#UNK) is not in the value set http://hl7.org/fhir/us/core/ValueSet/omb-ethnicity-category, and a code is required from this value set. (error message = Not in value set http://hl7.org/fhir/us/core/ValueSet/omb-ethnicity-category)
```

If there are any errors other than the following known error, please resolve them.

Known FHIR error:

Error @
 Bundle.entry[0].resource.ofType(Patient).extension[1].extension[0].value.ofType(Coding) (line 51, col83): The Coding provided (http://terminology.hl7.org/CodeSystem/v3-NullFlavor#UNK) is not in the value set http://hl7.org/fhir/us/core/ValueSet/omb-ethnicity-category, and a code is required from this value set. (error message = Not in value set http://hl7.org/fhir/us/core/ValueSet/omb-ethnicity-category)

Error Handling

Error messaging largely follows standard HTTP response codes where 4xx errors indicate a problem with the caller (transplant center) and 5xx errors indicate a problem with the service (NMDP).

400	Bad request (malformed, incomplete, cannot be processed)
401 / 403	Authorization issues (failed authentication, unauthorized access)
405	Method not allowed (POST=create, PUT=update, GET=health)
429	Too many requests sent within a certain timeframe.
500	Internal server error (something went wrong on the server)
503	Service is not available
504	Gateway timeout (response took too long)

If there are validation errors related to your request, you'll receive a "400 Bad Request" response along with a description of the error. Below are examples of 4xx and 5xx responses:

When TC ID is missing from the payload	HTTP 400/Bad Request	application/json	{ "code": 400, "message": "Patient does not have the minimum data needed to register in MatchSource, please provide data for [Organization identifier.value]" }
Patient-Import unable to find patient	HTTP 400/Bad request	application/json	{ "code": 400, "message": "Patient does not exist in MatchSource for RID: 1234569" }
When patient is not found for given input	HTTP 400/Bad Request	application/json	{ "code": 400, "message": "Patient does not exist in matchsource for provided rid, tcId and birthDate" }
When incorrect patient ID sent	HTTP 400/Bad Request	application/json	{ "code": 400, "message": "Invalid Patient RID." }
When TC sent invalid values in payload	HTTP 400/Bad Request	application/json	{ "code": 400, "message": "Patient data does not meet requirements to register in MatchSource, please provide correct data for [Patient communication.language.coding.system," }

			<pre>Patient communication.language.coding.code]" }</pre>
When TC did not send required values in payload	HTTP 400/Bad Request	application/json	<pre>{ "code": 400, "message": "Patient does not have the minimum data needed to register in MatchSource, please provide data for [Patient name.family, Patient birthDate]" }</pre>
When TC sent invalid values and did not send required values in payload	HTTP 400/Bad Request	application/json	<pre>{ "code": 400, "message": "Patient does not have the minimum data needed to register in MatchSource, please provide data for [Patient birthDate]Patient data does not meet requirements to register in MatchSource, please provide correct data for [Patient communication.language.coding.system, Patient communication.language.coding.code]" }</pre>
When TC has more than one instance of an observation in payload	HTTP 400/Bad Request	application/json	<pre>{ "code": 400, "message": "Patient data does not meet requirements to register in MatchSource, Multiple Observations provided for [PatientWeight]" }</pre>
When TC sent invalid LOINC for HLA	HTTP 400/Bad Request	application/json	<pre>{ "code": 400, "message": "Unknown HLA Observation(s) found [Observation contains Invalid HLA Loinc Code(s): [Invalid-Loinc-Code-Value]]" }</pre>
TC ID not matching TC that patient is tied to	HTTP 401/Unauthorized	application/json	<pre>{ "code": 401, "message": "Incomplete security requirements." }</pre>
Throttle limit reached	HTTP 429/ Too many requests	application/json	<pre>{</pre>

			<pre>"code": 429, "message": "TC: <tc id> has reached its Call Limit of 1 Call(s) within <x amount of> Minute(s). Calls will remain Throttled until: <Date and time>." }</pre>						
Potential Duplicate(s) Patient Found	HTTP 500/Internal server error	application/json	<pre>{ "duplicatePatients": [{ "nmdpId": "1234569", "localId": "someLocalId", "status": "IN_PROGRESS", "isAssignedToTc": true }] }</pre> <table><tr><th>Field Name</th><th>Type</th><th>Description</th></tr><tr><td>isAssignedToTc</td><td>boolean</td><td>true: Patient is assigned to the calling TC. false: Patient is not assigned to calling TC. Please contact your case manager if you have additional questions.</td></tr></table>	Field Name	Type	Description	isAssignedToTc	boolean	true: Patient is assigned to the calling TC. false: Patient is not assigned to calling TC. Please contact your case manager if you have additional questions.
Field Name	Type	Description							
isAssignedToTc	boolean	true: Patient is assigned to the calling TC. false: Patient is not assigned to calling TC. Please contact your case manager if you have additional questions.							
MatchSource is unavailable	HTTP 503/Service unavailable	application/json	<pre>{ "code": 503, "message": "System(s) are unavailable at this time please try again later. If issue persists please reach out to technical team." }</pre>						

A successful *create* call will return a “200 OK” response and include the generated patient ID:

```
{
  "id": "5676401"
}
```

This NMDP ID should be captured and stored by the caller to provide a mapping between the EHR system record (e.g. Epic) and the NMDP patient record. This ID is required for calls to the patient “*update*” feature.

Helpful Links

Patient-Import **Test** Environment

- **Version 3.0**

FHIR IG:	https://fhir.nmdp.org/ig/matchsync/index.html
Sample Payload:	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/example/patient-payload-example.json
Swagger:	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/swagger-ui/index.html
API spec:	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/api-docs
Service “create” endpoint (POST):	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/patient
Service “update” endpoint (PUT):	https://int-api.nmdp.org/int-matchsync-patient-import/3.0/patient/{RID}

Patient-Import **Production** Environment

- **Version 3.0**

FHIR IG:	https://fhir.nmdp.org/ig/matchsync/index.html
Sample Payload:	https://api.nmdp.org/p-matchsync-patient-import/3.0/example/patient-payload-example.json
Swagger:	https://api.nmdp.org/p-matchsync-patient-import/3.0/swagger-ui/index.html
API spec:	https://api.nmdp.org/p-matchsync-patient-import/3.0/api-docs
Service “create” endpoint (POST):	https://api.nmdp.org/p-matchsync-patient-import/3.0/patient
Service “update” endpoint (PUT):	https://api.nmdp.org/p-matchsync-patient-import/3.0/patient/{RID}

MatchSource[®] (application login):

- Test environment: <https://imatchsource-corebiz.nmdp.org/#/login>
- Production environment: <https://matchsource-core.nmdp.org/#/login>

HL7 FHIR

- <https://fhir.org/>
- <https://hapifhir.io/> (Java-tooling for FHIR and FHIR validation)
 - https://hapifhir.io/hapi-fhir/docs/getting_started/introduction.html
 - <https://github.com/hapifhir/hapi-fhir> (Spring support)
- <https://fhir.nmdp.org/> (NMDP FHIR space)
- <http://loinc.org> (LOINC coding system)

Tools

- <https://www.postman.com/> (Test tool for RESTful services)

Addendum

Sample: Minimum Required for a Successful “In-Progress” Create

```
{
  "resourceType": "Bundle",
  "meta": {
    "profile": ["http://fhir.nmdp.org/ig/matchsource/StructureDefinition/msbundle"],
    "security": [
      {
        "system": "http://terminology.nmdp.org/codesystem/transplant-center",
        "code": "tc_123"
      }
    ]
  },
  "type": "collection",
  "timestamp": "2024-02-14T21:30:20-05:00",
  "entry": [
    {
      "fullUrl": "urn:uuid:cc111e78-15eb-430d-b337-a4418494bedc",
      "resource": {
        "resourceType": "Patient",
        "meta": {
          "profile": ["http://fhir.nmdp.org/ig/matchsource/StructureDefinition/mspatient"],
          "security": [
            {
              "system": "http://terminology.nmdp.org/codesystem/transplant-center",
              "code": "tc_123"
            }
          ]
        },
        "name": [
          {
            "family": "Muntz",
            "given": [
              "Nelson",
              "B"
            ]
          }
        ],
        "gender": "male",
        "birthDate": "1976-03-09",
        "identifier": [
          {
            "use": "usual",
            "system": "http://terminology.nmdp.org/identifier/local-123",
            "value": "local3EpicId3003"
          }
        ]
      }
    }
  ]
}
```

```

    ],
    "communication": [
      {
        "language": { "coding": [
          {
            "system": "urn:ietf:bcp:47",
            "code": "ENG",
            "display": "English"
          }
        ] },
        "preferred": true
      }
    ],
    "managingOrganization": {
      "reference": "urn:uuid:413d5929-1fc5-43a4-ad48-e7ffe6af768c"
    }
  },
  {
    "fullUrl": "urn:uuid:413d5929-1fc5-43a4-ad48-e7ffe6af768c",
    "resource": {
      "resourceType": "Organization",
      "meta": {
        "profile":
["http://fhir.nmdp.org/ig/matchsource/StructureDefinition/transplantcenter"],
        "security": [
          {
            "system": "http://terminology.nmdp.org/codesystem/transplant-center",
            "code": "tc_123"
          }
        ]
      },
      "identifier": [
        {
          "use": "usual",
          "system": "http://terminology.nmdp.org/identifier/transplantcenter",
          "value": "123"
        }
      ],
      "name": "<Human Readable Transplant Center Name>"
    }
  }
]
}

```