

**NRDR**

**GRID Data Exchange Specification**

Version 1.4

September 12, 2023

Table of Contents

[1 Revision History 2](#_Toc138170222)

[2 GRID Data Elements 3](#_Toc138170223)

[2.1 GRID Exam Data Submission 3](#_Toc138170224)

[2.2 Data Model 4](#_Toc138170225)

[3 JSON Schema 6](#_Toc138170226)

[3.1 GRID Exam 2.1 JSON Schema 6](#_Toc138170227)

[3.2 GRID JSON Mapping Specification 11](#_Toc138170228)

[3.3 NRDR Web Service Status Schema 12](#_Toc138170229)

[3.3.1 Transaction Statuses 13](#_Toc138170230)

[3.4 NRDR Web Service Error Response Schema 13](#_Toc138170231)

[4 NRDR Transaction API 15](#_Toc138170232)

[4.1 Test Environment 15](#_Toc138170233)

[4.2 Authentication 15](#_Toc138170234)

[4.3 Create, Update or Replace Transaction 15](#_Toc138170235)

[4.3.1 Examples POST 16](#_Toc138170236)

[4.4 Get Transaction Status 19](#_Toc138170237)

[4.4.1 Examples GET 19](#_Toc138170238)

[5 Validation & Error Messages 21](#_Toc138170239)

NRDR GRID Exam Data Exchange

# Revision History

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Description** |
| 9/28/2014 | 1.0 | Draft |
| 12/4/2014 |  | Added list of web service’s methods with description. Added samples of the web service usage. |
| 1/20/2015 |  | Added UAT environment information. |
| 1/23/2015 |  | Updated the NON PQRS data element reference spreadsheet to Non-PQRS-GRID-Data-Elements-20150112.xlxs |
| 1/29/2015 |  | Updated API section with the refresh token information |
| 4/26/2015 |  | Included ACR SSO flow and edits |
| 11/22/2015 |  | Corrected the name of the following parameters of the GET method:  Updated transaction\_id to id  Updated facility\_id to facilityid |
| 10/11/2018 |  | Added error codes G1012, G2057, G2091, G2092, G3003, G3042, G3014, G3110, G3111, G3112, G3113, G3114, G3115, G3116, G3117, G3118, G3119, G3120, G3121, G3122, G3123, G3124  The following error codes are obsolete: G3082, G3072, G3061, G3032, G3092, and G3094 |
| 02/19/2019 |  | Added error codes G3032, G3033, G3045, G3072 |
| 04/17/2019 |  | Added new data elements to JSON schema: Multiple\_Readers, Place\_of\_Service  New error codes added: G3095, G3096 |
| 09/17/2019 |  | Data model updated. JSON 2.0 schema added  New error codes added: G3004, G3005, G3006, G3007, G3009, G3010, G3131 |
| 09/17/2019 |  | Error message updated for error code G3045 |
| 10/09/2019 |  | New error codes added: G3211, G3212, G3221, G3222 |
| 08/27/2020 |  | Added description for new JSON schema ‘2.1’ |
| 08/27/2020 |  | New error codes added: G3230, G3231 |
| 04/01/2021 |  | GRID JSON 1.2 schema retired |
| 05/15/2023 | 1.4 | GRID 2.0 version retired |
| 06/01/2023 | 1.4 | Added API response statuses definitions and examples |
|  |  |  |

# GRID Data Elements

This document describes the GRID Exam data elements, JSON schema, NRDR GRID APIs and error messages. It also demonstrates how to use the authorization token when consuming the GRID APIs.

## GRID Exam Data Submission

The NRDR provides a set of APIs which allows authorized partners to submit GRID exam data for their affiliated facilities. Partner is authenticated via the ACR SSO Authentication Service. Once authenticated, ACR SSO will provide access token to partner to submit GRID Exam data and query submission status using the NRDR GRID APIs.

A diagram of a grid api

Description automatically generated with low confidence

The diagram below depicts the events that take place in the ACR SSO during the authentication and authorization processes. Please refer to the [*ACR SSO Authentication*](https://nrdrsupport.acr.org/support/solutions/articles/11000090513) *article* for a detailed description of ACR SSO service.

A picture containing text, diagram, screenshot, line

Description automatically generated

The NRDR GRID APIs are a set of REST web services. They return an acknowledgement to the partner when submission is received. It is the responsibility of the partner to capture this acknowledgement to ensure the submission has reached ACR SSO successfully.

The frequency of the data submission is defined by the partner, however, the ACR prefers the submission to happen more frequent so that the facility/physician will have a faster turnaround time for review and be able to resubmit data before the CMS submission deadline if needed.

Partners shall be able to query the status of the submitted transactions by using NRDR GRID API.

## Data Model

The GRID Exam data to be submitted to the ACR NRDR shall be “wrapped” in a transaction. This approach provides the flexibility of submitting multiple exams for a facility within a single transaction. Note: each transaction shall contain exams for one facility.



Where K>=1, L=5, M=5, N=3.

The following composite key shall be used to identify unique GRID exam record:

* Facility ID
* Exam Unique ID
* CPT HCPCS Code

For each exam in a transaction the API will base on its composite key to determine if it should create a new exam record or update an existing exam record in the database. If the composite key is found in the database, the API will update the existing record; otherwise, it will create a new exam record.

# JSON Schema

## GRID Exam 2.1 JSON Schema

This schema describes the data elements in a transaction for GRID file version 2.1:

{

"name": "GRID\_Exam\_Transaction",

"Properties": {

"Transaction\_ID": {

"type": "string",

"required": true

},

"Transaction\_DateTime": {

"type": "string",

"required": true

},

"Num\_of\_Exam\_Included": {

"type": "number",

"required": true

},

"Facility\_ID": {

"type": "string",

"required": true

},

"PartnerID": {

"type": "string",

"required": true

},

"AppID": {

"type": "string",

"required": true

},

"Previous\_Transaction\_ID": {

"type": "string",

"required": false

},

"GRID\_File\_Version\_Num": {

"type": "string",

"required": false

},

"Exam": {

"type": "array",

"required": true,

"properties": {

"Exam\_Unique\_ID": {

"type": "string",

"required": true

},

"Physician\_NPI": {

"type": "string",

"required": true

},

"Physician\_Local\_ID": {

"type": "string",

"required": false

},

"Multiple\_Readers": {

"type": "number",

"required": true

},

"Patient\_Age": {

"type": "number",

"required": true

},

"Patient\_Gender": {

"type": "string",

"required": true

},

"Study\_Name": {

"type": "string",

"required": false

},

"CPT\_HCPCS\_Code": {

"type": "string",

"required": false

},

"ICD\_10\_Procedure\_Code": {

"type": "string",

"required": false

},

"Modality\_Procedure": {

"type": "number",

"required": false

},

"Place\_of\_Service": {

"type": "number",

"required": true

},

"Breast\_Ultrasound": {

"type": "string",

"required": false

},

"Date\_Exam\_Completed": {

"type": "string",

"required": true

},

"Time\_Exam\_Completed": {

"type": "string",

"required": true

},

"Date\_Final\_Report\_Signed": {

"type": "string",

"required": true

},

"Time\_Final\_Report\_Signed": {

"type": "string",

"required": true

},

"Extravasation\_Occurred": {

"type": "string",

"required": false

},

"Height": {

"type": "string",

"required": false

},

"Weight": {

"type": "string",

"required": false

},

"BMI": {

"type": "string",

"required": false

},

"Medical\_History": {

"type": "string",

"required": false

},

"Procedure\_History": {

"type": "string",

"required": false

},

"Smoking\_Status": {

"type": "string",

"required": false

},

"Number\_of\_Years\_Since\_Quit": {

"type": "string",

"required": false

},

"Number\_Pack\_Years": {

"type": "string",

"required": false

},

"Body\_Region": {

"type": "string",

"required": false

},

"Anatomy": {

"type": "string",

"required": false

},

"Clinical\_Focus": {

"type": "string",

"required": false

},

"Use\_of\_Contrast": {

"type": "string",

"required": false

},

"Modality\_Modifier": {

"type": "string",

"required": false

},

"Procedure\_Modifier": {

"type": "string",

"required": false

},

"Nuclear\_Agent": {

"type": "string",

"required": false

},

"Final\_Report\_Findings ": {

"type": "string",

"required": false

},

"FDG\_PET\_Measurements\_Documented": {

"type": "string",

"required": false

},

"PE\_Documentation": {

"type": "string",

"required": false

},

"Structured\_Scoring\_System\_Method": {

"type": "string",

"required": false

},

"Incidental\_Findings": {

"type": "string",

"required": false

},

"Incidental\_Mass": {

"type": "array",

"required": false,

"properties": {

"Incidental\_Mass\_Type": {

"type": "string",

"required": false

},

"Incidental\_Mass\_Size": {

"type": "string",

"required": false

},

"Incidental\_Mass\_Impression": {

"type": "string",

"required": false

},

}

},

"Followup\_Imaging\_Recommendations": {

"type": "string",

"required": false

},

"Recommended\_Followup\_Imaging": {

"type": "array",

"required": false,

"properties": {

"Recommended\_Imaging\_Modality": {

"type": "string",

"required": false

},

"Recommended\_Imaging\_Time": {

"type": "string",

"required": false

},

"Recommendation\_Source\_Documented": {

"type": "string",

"required": false

},

}

},

"Followup\_Procedure\_Recommendations": {

"type": "string",

"required": false

},

"Recommended\_Followup\_Procedure": {

"type": "array",

"required": false,

"properties": {

"Recommended\_Procedure\_Type": {

"type": "string",

"required": false

},

"Recommended\_Procedure\_Anatomy": {

"type": "string",

"required": false

},

"Recommended\_Procedure\_Time": {

"type": "string",

"required": false

},

"Recommendation\_Source\_Documented": {

"type": "string",

"required": false

},

"TIN": {

"type": "string",

"required": false

},

}

}

}

}

}

}

## GRID JSON Mapping Specification

The *GRID\_JSON\_Mapping\_<version>.xlsx* document describes the validation rules and format of the JSON data elements. Most of the values of the data elements are mapped to either numeric or alpha-numeric codes.

To have a “completed” status on an exam record, you must submit the data elements marked as “Y” or “Conditional” under the “Required to Complete Exam” column.

The data elements that are marked as “Optional” are not required for submission, a record will not be rejected if an optional field is missing. However, if you do know them, we expect you to provide them. This expectation of complete data is specified in the participation agreement signed by all participating facilities.

## NRDR Web Service Status Schema

NRDR web service transaction status schema describes the data elements and the validation statuses of a transaction, and the exams associated with it.

You will use the GET request to query the error encountered during the submission process.

Note: the value of the **Error\_Msg** attribute consists of multiple error messages that occurred while validating a transaction. Each message contains the error code-message pair. Messages are separated by the ‘|’ delimiter.

For example:

G2001:Missing Transaction\_ID|G2011:Missing Transaction\_DateTime

GRID Exam Transaction Status schema:

{

"name": "GRID\_Exam\_Transaction\_Status",

"Properties": {

"Transaction\_ID": {

"type": "string",

"required": true

},

"Transaction\_DateTime": {

"type": "string",

"required": true

},

"Num\_of\_Exam\_Included": {

"type": "number",

"required": true

},

"Num\_of\_Exam\_In\_Error": {

"type": "number",

"required": true

},

"PartnerID": {

"type": "string",

"required": true

},

"AppID": {

"type": "string",

"required": true

},

"Facility\_ID": {

"type": "string",

"required": true

},

"Status": {

"type": "string",

"required": true

},

"Error\_Msg": {

"type": "string",

"required": false

},

"Exam\_Errors": {

"type": "array",

"required": true,

"properties": {

"Exam\_Unique\_ID”: {

“type”: “string”,

“required”: true

},

"Error\_Msg": {

"type": "string",

"required": true

}

}

}

}

}

### Transaction Statuses

The following transaction status is possible:

* **Enqueued** – Transaction has been placed in queue and awaiting to be loaded
* **Received** – Transaction has been received and is pending for validation
* **ValidationFail** – Nothing got loaded. Failed at exam processing
* **ValidationSuccessful** – Validation has complete, pending for loading into database
* **Loading** – Data is loading, transaction is being created
* **Posted** – The entire transaction is loaded successfully
* **UpdatedWithNewTransaction** – This transaction is replaced with another transaction
* **PartiallySuccessful** – Some elements are loaded and some are rejected
* **Failed** – Transaction failed to be created due to an error

## NRDR Web Service Error Response Schema

NRDR transaction web service error responseschema describes detailed error information which is returned by all API responses with a status code other than 200:

GRID\_WebService\_Error schema:

{

"name": "GRID\_WebService\_Error",

"Properties": {

"error\_code": {

"type": "string",

"required": true

},

"error\_description": {

"type": "string",

"required": true

}

}

# NRDR Transaction API

## Test Environment

Base URL: <https://nrdr9x.acr.org/NonPQRSapi/grid/transactions>

Application ID: 10005

Partner ID: to be provided

Facility ID: to be provided

## Authentication

All API requests must be authenticated and authorized. To call the API you must add JWT token received from ACR SSO to the “Authorization” request header using the following format:

Bearer <jwt-token>

The 401 status code indicates errors related to the authentication. Following error codes are used to provide more detailed error:

* ‘invalid\_token’, token is invalid or expired
* ‘missing\_token’, token is not provided in the request header

## Create, Update or Replace Transaction

**POST**

POST request is used to submit a new set or update GRID exams data in the GRID\_Exam\_Transaction schema. Each transaction must have a unique Transaction\_ID. Partner is required to maintain the uniqueness of the Transaction\_ID.

To update the existing exam, the same composite key must be provided (including patient identifiers). When the existing exam record is updated, not-null existing values will not be replaced with null values from newly submitted transaction.

The Previous\_Transaction\_ID is used for resubmission. If you need to replace a previous transaction entirely, you will submit a new transaction and populate the Previous\_Transaction\_ID with the Transaction\_ID that was submitted previously. In this case, exam(s) of previous transaction will be deleted entirely from the database and exam(s) from new transaction will be loaded.

Please note, submissions take up to 72 hours to process and load to NRDR database.

Acceptable request representations:

* application/json

Possible responses:

* 200, returned when transaction data is uploaded and transaction is created successfully,
* 400, content-type: application/json
  + error\_code = ‘data\_error’

Transaction data is missing or doesn’t have valid JSON representation

* 403, content-type: application/json
  + error\_code = ‘no\_permission’

Authenticated user doesn’t have permissions to upload transaction data for a given facility

### Examples POST

Sample parameters:

Facility\_ID: "100000"

PartnerID: "10000"

AppID: "10005"

Authorization: “Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJodHRwOi8vd3d3LnczLm9yZy8yMDAwLzA5L3htbGRzaWcjcnNhLXNoYTEiLCJ4NXQiOiIzWndDNC15STBQQmNHeDAyWlNsQjdaa1NRLUUifQ.eyJpc3MiOiJodHRwczovL1NlY3VyZUF1dGgwMVZNLmFjci5vcmciLCJhdWQiOiJodHRwczovL1NlY3VyZUF1dGgwMVZNLmFjci5vcmciLCJuYmYiOjE0MjIyNzY3ODIsImV4cCI6MTQyMjM2MzE4MiwiY2xpZW50X2lkIjoiMWY1ZjM5NTI0ZjIyNGRmMDg0NTIwYTJmYWE5YTkyNzUiLCJzY29wZSI6WyJvcGVuaWQiLCJwcXJzIl0sInN1YiI6IjEwMDAwMF9udWFuY2UiLCJhbXIiOiJwYXNzd29yZCIsImF1dGhfdGltZSI6IjE0MjIyNzY3NTciLCJpZHAiOiJTZWN1cmVBdXRoNSJ9.AFsoPNtIT8N0N6w1y0OZLokdplfRM2Y8OAvquMXpkoI\_6b9ANEhXY72ubuylt6-4vdJ165aVUXPRDTF37UT9cqv21hRTpOfYYM0tV5gksUbvLzKaPFt6EWbB8ctQnHs1pxyDGSbKDM-dR6upj6x0eZ7O46Zwi8o8Zhkeu0bFZPNmLS7y96\_A9uyGBkbt69Ih0aCBNRvtabqkD3vJsWB4EIfiqJJ2RO8XPCGfSoYAnuosgH5Cm6kMNN7WiX4rgFps20MryBLtutKztcNhaj-l65gzxyrJpgmq8r9dBWiFI6nFc-rKkNr6Gdj2TH9EABloC6aO\_CQQFtgrXFLEchnSPw”

**Note**: the authentication/authorization mechanism in the GRID Exam API adopts the ACR SSO security model which requires an access token after authenticated successfully. The token in the sample parameters above is merely an example.

function createTransaction(transactionData, token) {

    $.ajax({

        type: 'POST',

        url: "https://nrdr9x.acr.org/NonPQRSapi/grid/transactions",

        data: JSON.stringify(transactionData),

        dataType: "json",

        contentType:"application/json; charset=utf-8",

        headers: { "Authorization": "Bearer " + token },

        success: function (data) {

            // handle data load success

        }

    });

}

This function can be used in the following way:

var transactionData =

{

"Transaction\_ID": "g-2503-04",

"Transaction\_DateTime": "04/09/2023",

"Num\_of\_Exam\_Included": 3,

"Facility\_ID": "100000",

"PartnerID": "40001",

"AppID": "20002",

"Previous\_Transaction\_ID": null,

"GRID\_File\_Version\_Num": "2.1",

"Exam": [

{

"Exam\_Unique\_ID": "OPA-002",

"Patient\_Age": "25",

"Patient\_Gender": "M",

"Study\_Name": "study name",

"Physician\_NPI": "123456789",

"Physician\_Local\_ID": "755",

"Multiple\_Readers": "0",

"CPT\_HCPCS\_Code": "99397",

"ICD\_10\_Procedure\_Code": "ICD10",

"Modality\_Procedure": "1",

"Place\_of\_Service": "1",

"Breast\_Ultrasound": null,

"Date\_Exam\_Completed": "01/01/2020",

"Time\_Exam\_Completed": "15:35:23",

"Date\_Final\_Report\_Signed": "01/01/2020",

"Time\_Final\_Report\_Signed": "16:21:33",

"Extravasation\_Occurred": "Y",

"Height": "99.99",

"Weight": "99.99",

"BMI":"99.99",

"Medical\_History": "0",

"Procedure\_History": "0",

"Smoking\_Status":"1",

"Number\_of\_Years\_Since\_Quit":null,

"Number\_Pack\_Years":"999.9",

"Body\_Region":"88",

"Anatomy":"1",

"Clinical\_Focus":"1",

"Use\_of\_Contrast": "1",

"Modality\_Modifier":"1",

"Procedure\_Modifier":"20",

"Nuclear\_Agent": "10",

"Final\_Report\_Findings": "1",

"FDG\_PET\_Measurements\_Documented": "1",

"PE\_Documentation": "1",

"Structured\_Scoring\_System\_Method":"1",

"Incidental\_Findings":"1",

"Incidental\_Mass":[

{

"Incidental\_Mass\_Type":1,

"Incidental\_Mass\_Size":"9999.9",

"Incidental\_Mass\_Impression":"0"

},

{

"Incidental\_Mass\_Type": "1",

"Incidental\_Mass\_Size":"9999.9",

"Incidental\_Mass\_Impression":"0"

},

{

"Incidental\_Mass\_Type": "1",

"Incidental\_Mass\_Size":"9999.9",

"Incidental\_Mass\_Impression":"0"

},

{

"Incidental\_Mass\_Type": "1",

"Incidental\_Mass\_Size":"9999.9",

"Incidental\_Mass\_Impression":"0"

},

{

"Incidental\_Mass\_Type": "1",

"Incidental\_Mass\_Size":"9999.9",

"Incidental\_Mass\_Impression":"0"

}

],

"Followup\_Imaging\_Recommendations":"1",

"Recommended\_Followup\_Imaging":[

{

"Recommended\_Imaging\_Modality":1,

"Recommended\_Imaging\_Time":9999,

"Recommendation\_Source\_Documented":0

},

{

"Recommended\_Imaging\_Modality":1,

"Recommended\_Imaging\_Time":9999,

"Recommendation\_Source\_Documented":0

},

{

"Recommended\_Imaging\_Modality":1,

"Recommended\_Imaging\_Time":9999,

"Recommendation\_Source\_Documented":0

},

{

"Recommended\_Imaging\_Modality":1,

"Recommended\_Imaging\_Time":9999,

"Recommendation\_Source\_Documented":0

},

{

"Recommended\_Imaging\_Modality":1,

"Recommended\_Imaging\_Time":9999,

"Recommendation\_Source\_Documented":0

}

],

"Followup\_Procedure\_Recommendations":1,

"Recommended\_Followup\_Procedure":[

{

"Recommended\_Procedure\_Type":5,

"Recommended\_Procedure\_Anatomy":1,

"Recommended\_Procedure\_Time":9999,

"Recommendation\_Source\_Documented":0

},

{

"Recommended\_Procedure\_Type":5,

"Recommended\_Procedure\_Anatomy":1,

"Recommended\_Procedure\_Time":9999,

"Recommendation\_Source\_Documented":0

},

{

"Recommended\_Procedure\_Type":5,

"Recommended\_Procedure\_Anatomy":1,

"Recommended\_Procedure\_Time":9999,

"Recommendation\_Source\_Documented":

}

],

"TIN": null

} ,

]

}

// get ACR SSO token

var token = GetAuthorizationToken();

createTransaction(transactionData, token);

## Get Transaction Status

**GET**

GET request is used to get a transaction JSON object for a single transaction.

Possible responses:

* 200, content-type: application/json

Response will contain JSON data according to NRDR web service status schema

* 400, content-type: application/json
  + error\_code = ‘missing\_param’

Indicates a missing transaction ID parameter

* 404, content-type: application/json
  + error\_code = ‘invalid\_param’

Indicates that transaction with a given ID doesn’t exist

### Examples GET

Sample parameters:

id: “26”     (transaction id)

FacilityID: "100000"

               PartnerID: "30004"

               AppID: "10005"

Authorization: “Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJodHRwOi8vd3d3LnczLm9yZy8yMDAwLzA5L3htbGRzaWcjcnNhLXNoYTEiLCJ4NXQiOiIzWndDNC15STBQQmNHeDAyWlNsQjdaa1NRLUUifQ.eyJpc3MiOiJodHRwczovL1NlY3VyZUF1dGgwMVZNLmFjci5vcmciLCJhdWQiOiJodHRwczovL1NlY3VyZUF1dGgwMVZNLmFjci5vcmciLCJuYmYiOjE0MjIyNzY3ODIsImV4cCI6MTQyMjM2MzE4MiwiY2xpZW50X2lkIjoiMWY1ZjM5NTI0ZjIyNGRmMDg0NTIwYTJmYWE5YTkyNzUiLCJzY29wZSI6WyJvcGVuaWQiLCJwcXJzIl0sInN1YiI6IjEwMDAwMF9udWFuY2UiLCJhbXIiOiJwYXNzd29yZCIsImF1dGhfdGltZSI6IjE0MjIyNzY3NTciLCJpZHAiOiJTZWN1cmVBdXRoNSJ9.AFsoPNtIT8N0N6w1y0OZLokdplfRM2Y8OAvquMXpkoI\_6b9ANEhXY72ubuylt6-4vdJ165aVUXPRDTF37UT9cqv21hRTpOfYYM0tV5gksUbvLzKaPFt6EWbB8ctQnHs1pxyDGSbKDM-dR6upj6x0eZ7O46Zwi8o8Zhkeu0bFZPNmLS7y96\_A9uyGBkbt69Ih0aCBNRvtabqkD3vJsWB4EIfiqJJ2RO8XPCGfSoYAnuosgH5Cm6kMNN7WiX4rgFps20MryBLtutKztcNhaj-l65gzxyrJpgmq8r9dBWiFI6nFc-rKkNr6Gdj2TH9EABloC6aO\_CQQFtgrXFLEchnSPw”

**Note**: the authentication/authorization mechanism in the GRID Exam API adopts the ACR SSO security model which requires an access token after authenticated successfully. The token in the sample parameters above is merely an example.

function getTransaction(transactionId, token, partnerId, appId, facilityId) {

    $.ajax({

        type: 'GET',

        url: "https://nrdr9x.acr.org/NonPQRSapi/grid/transactions",

data: { "id": transactionId, "appid": appId, "partnerid":

partnerId,"facilityid": facilityId },

        headers: { "Authorization": "Bearer " + token },

        success: function (data) {

            // handle JSON with transaction data

        }

    });

}

JSON response example:   
{

"Transaction\_ID": "26",

"Transaction\_DateTime": "2023-02-02T00:00:00",

"Num\_of\_Exam\_Included": 1,

"Num\_of\_Exam\_In\_Error": 0,

"PartnerID": 30004,

"AppID": 10005,

"Facility\_ID": "100000",

"Error\_Msg": "",

"Exam\_Errors": [

{

"Exam\_Unique\_ID": "uniq0001",

"Error\_Msg": "G3045:Warning. Physician NPI 0000000000 does not exist for this facility ID/registry"

}

],

"Status": "Posted"

}

# Validation & Error Messages

The Validator will validate the following:

1. JSON syntax
2. Required fields

The table below lists the possible messages generated by the validator during the validation process for a transaction:

| **Error Code** | **Message** |
| --- | --- |
| **Transaction level** | |
| G0000 | Transaction is loaded successfully |
| G0001 | Transaction has been received and is pending for validation |
| G0002 | Validation has been completed, pending for loading to database |
| G0003 | Transaction was partially loaded. Some elements did not pass validation and were rejected. |
| G0004 | The entire transaction was rejected due to failure in validation |
| G0005 | This transaction has been replaced with another transaction |
| G1003 | Unable to import data submitted. Data does not match schema |
| G1004 | Unable to import data submitted. Data is empty or null |
| G1005 | Error importing data <exception message> |
| G1010 | User account is not authorized to submit data to GRID |
| G1012 | Partner <Partner ID> is not authorized to submit data for facility <facility ID> and/or application <application> |
| G2001 | Missing Transaction\_ID |
| G2002 | Invalid Transaction\_ID. Maximum length is 50 characters |
| G2011 | Missing Transaction\_DateTime |
| G2012 | Transaction\_DateTime is not a valid date time format |
| G2013 | Transaction\_DateTime is a future date |
| G2014 | Transaction\_ID must be unique. <transaction\_id> has been submitted in a previous transaction |
| G2021 | Missing Num\_of\_Exam\_Included |
| G2022 | Num\_of\_Exam\_Included is not an integer |
| G2031 | Missing PartnerID |
| G2032 | Invalid PartnerID |
| G2041 | Missing AppID |
| G2042 | Invalid AppID |
| G2051 | Missing Facility\_ID |
| G2052 | Invalid Facility\_ID |
| G2053 | Facility has not registered for GRID. Submission aborted |
| G2054 | Facility has not been accepted to GRID. Submission aborted |
| G2057 | Invalid FacilityID. User has no access to the registry of the facility |
| G2062 | Invalid Prev\_Transaction\_ID. No transactions with Transaction ID <prev\_transaction\_id> was received till now |
| G2071 | Missing exam object information |
| G2072 | Data in tag Num\_of\_Exams\_Included <num\_of\_exams\_included>does not match with exam received <total # exam received> |
| G2081 | Some of the exams did not pass validation and were rejected |
| G2091 | Transaction has unknown element(s): <unknown\_elements\_list> |
| G2092 | Forbidden symbol ‘|’ in field(s): <fields\_names> |
| **Exam level** | |
| G3001 | Missing Exam Unique ID |
| G3003 | Invalid. Record - <number of fields> entries expected. <number of fields in the exam provided> read. |
| G3004 | Invalid {0}. Maximum array length is {1} elements |
| G3005 | Invalid {0}. Field must contain only the characters 0-9. Field may contain a decimal point |
| G3006 | Invalid {0}. Field must be numeric with up to {1} digits after the decimal point |
| G3007 | Invalid {0} format |
| G3009 | Invalid {0}. Possible values between {2} and {3} |
| G3010 | Invalid {0}. Values should be unique within array |
| G3081 | Missing Patient Gender |
| G3071 | Missing Patient Age |
| G3041 | Missing Physician\_NPI |
| G3042 | Invalid Physician\_NPI |
| G3011 | CPT\_HCPCS\_Code is required when neither ICD\_10\_Procedure\_Code nor Modality\_Procedure is provided |
| G3012 | Invalid CPT\_HCPCS\_Code |
| G3013 | Invalid CPT\_HCPCS\_Code. Field must contain only alphanumeric characters |
| G3014 | Invalid. CPT\_HCPCS\_Code length must be <number of characters> characters |
| G3021 | ICD\_10\_Procedure\_Code is required when neither CPT\_HCPCS\_Code nor Modality\_Procedure is provided |
| G3000 | Invalid. Missing Version Number |
| G3031 | Modality\_Procedure is required when neither CPT\_HCPCS\_Code nor ICD\_10\_Procedure\_Code is provided |
| G3032 | Invalid version number. Version number {0} not supported |
| G3033 | Invalid version number. Version number must be {0}. Field read: {1} |
|  |  |
| G3045 | Warning. Physician NPI does not exist for this facility ID/registry |
| G3051 | Breast\_Ultrasound is required when CPT\_HCPCS\_Code is not provided |
| G3061 | Missing Date\_Exam\_Completed |
| G3062 | Invalid {0} format. Valid format is mm/dd/yyyy |
| G3063 | Missing Time\_Exam\_Completed |
| G3064 | Invalid Time\_Exam\_Completed format: <time\_exam\_completed>. Valid format is hh:mi:ss in military format |
| G3072 | Invalid. Patient Age must be equal or less than 120 years |
| G3091 | Missing Date\_Final\_Report\_Signed |
| G3093 | Missing Time\_Final\_Report\_Signed |
| G3095 | Invalid. Missing Multiple\_Readers |
| G3096 | Invalid. Missing Place\_of\_Service |
| G3101 | Contrast\_Usage is required when CPT\_HCPCS\_Code is not provided |
| G3110 | Invalid <field\_name>. Maximum character length is <field\_max\_length> character(s) |
| G3111 | Invalid <field\_name>. Field must contain only the characters 0-9 |
| G3112 | Invalid <field\_name>. <Hour, Minutes or Seconds> must be in a <time\_format> format |
| G3113 | Invalid <field\_name>. <Hour, Minutes or Seconds> must be in a numeric <length> digit <format> format |
| G3114 | Invalid <field\_name>. <Hour, Minutes or Seconds> must be between <min\_value> and <max\_value> |
| G3115 | Invalid <field\_name>. Field must contain 'Y' or 'N' |
| G3116 | Invalid <field\_name>. Values <possible\_values> expected |
| G3117 | Invalid <field\_name>. Month must be in a 2 digit mm format |
| G3118 | Invalid <field\_name>. Month must be between 1 and 12 |
| G3119 | Invalid <field\_name>. Month must be a numeric value |
| G3120 | Invalid <field\_name>. Day must be in a 2 digit dd format |
| G3121 | Invalid <field\_name>. Day must be between 1 and 31 |
| G3122 | Invalid <field\_name>. Day must be a numeric value |
| G3123 | Invalid <field\_name>. Year must be in a 4 digit yyyy format |
| G3124 | Invalid <field\_name>. Year must be between 2014 and <current\_year> |
| G3125 | Invalid <field\_name>. Year must be a numeric value |
| G3126 | Invalid <field\_name>. Month must be before or on present date |
| G3127 | Invalid <field\_name>. Day must be before or on present date |
| G3128 | Invalid <field\_name>. Dates in a leap year in the month of February must have day values between 1 and 29 |
| G3129 | Invalid <field\_name>. Dates in the month of February must have day values between 1 and 28 |
| G3130 | Invalid <field\_name>. Dates in the months of April, June, September, and November must have day values between 1 and 30 |
| G3131 | Invalid <field\_name>. Incompatible set of values for multi selection field. |
| G3132 | Warning. Date&Time\_Report\_Signed must be on or after Date&Time\_Exam\_Completed |
| G3211 | Warning. Number\_Of\_Packs\_Year\_Smoking is not provided when Smoking status is ‘Current smoker’, ‘Former smoker’ or ’Smoker, current status unknown’ |
| G3212 | Invalid. Number\_Of\_Packs\_Year\_Smoking only applicable if Smoking status is ‘Current smoker’, ‘Former smoker’ or ‘Smoker, current status unknown’ |
| G3221 | Warning. Number\_Of\_Years\_Since\_Quit is not provided when Smoking status is ‘Former smoker’ |
| G3222 | Invalid. Number\_Of\_Years\_Since\_Quit only applicable if Smoking status is ‘Former smoker’ |
| G3230 | Invalid. TIN does not exist for this facility ID/registry/NPI |
| G3231 | Invalid TIN format. Must be 9-digits |
|  |  |