
ESP Data Extract Testing Guide

Revision History

Version Number	Modification Date	By	Description of Changes
1.0	22 May 2018	K. Eberhardt	Initial version
1.1	09 Apr 2019	K Eberhardt	Added Sections For Database Validation
1.2	11 Apr 2019	K. Eberhardt	Updated sections for database validation for lab results
1.3	24 Apr 2019	J. Miller	Minor updates after testing – fixed single quotes
1.4	29 Apr 2019	K. Eberhardt	Updated Deleting Test Data section to better warn against deletion of required provenances. Updates to formatting and layout. Updated TOC.
1.5	20 Jan 2020	K. Eberhardt J. Miller	Update queries. Add new queries to get load counts by years and months

Contents

1 Overview	3
2 Naming Files & Uploading To The ESP Server	4
3 Preparing File(s) For Processing	4
4 Loading The File(s)	4
5 Using the UI to Validate	6
6 Using the Database to Validate: Member	8
7 Using the Database to Validate: Provider	10
8 Using the Database to Validate: Visits	10
9 Using the Database to Validate: Medications	11
10 Using the Database to Validate: Lab Results	12
11 Using the Database to Validation: Historical Data Load	13
12 Troubleshooting Load Failures	13
13 Deleting Test Data	14

1 Overview

This document outlines a basic process for testing/validating client EMR data extract files. This process assumes sample extract files matching ESP file requirements have already been created.

This Document written for ESP version 3.4 and 3.5

2 Naming Files & Uploading To The ESP Server

1. Ensure the sample extract files match the naming convention outlined in the ESP FileSpec document.

	FILE NAMING CONVENTION:	EXAMPLE:
MEMBER	epicmem.esp.MMDDYYYY	epicmem.esp.03302014
PROVIDER	epicpro.esp.MMDDYYYY	epicpro.esp.12112013
VISIT	epicvis.esp.MMDDYYYY	epicvis.esp.09302015
MEDICATIONS	epicmed.esp.MMDDYYYY	epicmed.esp.12232015
LAB RESULTS	epicres.esp.MMDDYYYY	epicres.esp.03172017

2. Upload the files to the following directory on the ESP server:

```
/srv/esp/data/fake
```

3 Preparing File(s) For Processing

1. It is recommended to process 1 file at a time for testing. When possible, start with the provider file first followed by the member file and then any of the additional files.
2. Use the following commands to move the test file in to the proper directory for processing:

```
cd /srv/esp/data/fake  
mv <insert_filename_here> ../epic/incoming
```

For example:

```
cd /srv/esp/data/fake  
cp epicmed.esp.11292017 ../epic/incoming/
```

4 Loading The File(s)

1. Use the load_epic command to attempt the loading of the file(s):

```
cd /srv/esp/prod  
./bin/esp load_epic
```

2. When processing a particularly large file that may take a long time to complete, it is recommended to make the call using nohup. This will prevent the process from failing should you experience a network interruption while processing. This will log the output of the script to the nohup file.

```
cd /srv/esp/prod  
rm nohup.out  
nohup ./bin/esp load_epic &
```

```
tail -f nohup.out
```

3. Review the processing output. If errors are encountered that prevent the load from completing, see the Troubleshooting section below. The number of valid records loaded should match the number of rows in the file (assuming there are no duplicate natural keys in the data):

```
Valid records loaded:
```

```
{'epicall': 0,  
'epicgrd': 0,  
'epichpr': 0,  
'epicimm': 0,  
'epiclab': 0,  
'epiclnd': 0,  
'epicmad': 0,  
'epicmed': 0,  
'epicmem': 0,  
'epicmud': 0,  
'epicoex': 0,  
'epicoid': 0,  
'epicord': 0,  
'epicpid': 0,  
'epicpph': 0,  
'epicprb': 0,  
'epicprg': 0,  
'epicpro': 21,  
'epicres': 0,  
'epicsob': 0,  
'epicsoc': 0,  
'epicspc': 0,  
'epicvis': 0}
```

```
-----  
Errors:
```

```
{'epicall': 0,  
'epicgrd': 0,  
'epichpr': 0,  
'epicimm': 0,  
'epiclab': 0,  
'epiclnd': 0,  
'epicmad': 0,  
'epicmed': 0,  
'epicmem': 0,  
'epicmud': 0,  
'epicoex': 0,  
'epicoid': 0,  
'epicord': 0,  
'epicpid': 0,  
'epicpph': 0,  
'epicprb': 0,  
'epicprg': 0,  
'epicpro': 0,  
'epicres': 0,  
'epicsob': 0,  
'epicsoc': 0,  
'epicspc': 0,  
'epicvis': 0}
```

4. A load report is also written to the /srv/esp/data/load_reports directory:

```
cd /srv/esp/data/load_reports
ls -ltr (this will list the files by time modified. The most recent file will be listed at the
bottom)
view <insert_filename_here>
```

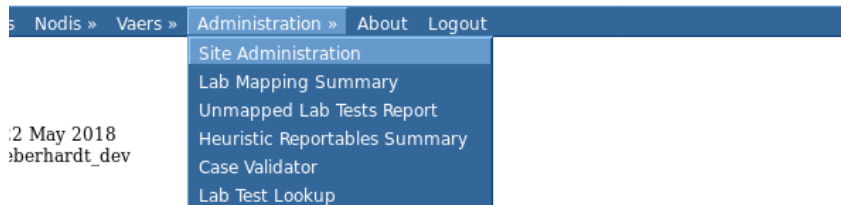
Sample load file contents:

```
+++++
Load summary for source file: epicpro.esp.08102017
Number of rows in source file: 21
Number of rows processed without error: 21
Number of rows processed with error: 0
+++++
Number of database rows created (inserted): 0
Number of updates performed on existing rows: 21
Number of rows in the EMR table having the current Provenance ID 390: 21
~
~
```

5 Using the UI to Validate

1. Log in to the ESP User Interface (UI).
2. Navigate to “Administration -> Site Administration”

ESP — ELECTRONIC MEDICAL RECORD SUPPORT





- In the “EMR” section click on the item name that matches the test file that was loaded:

Emr		
Allergys	+ Add	Change
Encounter type maps	+ Add	Change
Encounters	+ Add	Change
Hospital_problems	+ Add	Change
Immunizations	+ Add	Change
Lab Orders	+ Add	Change
Lab Specimen Observations	+ Add	Change
Lab Specimens	+ Add	Change
Lab Test Results	+ Add	Change
Lab infos	+ Add	Change
Order Extensions	+ Add	Change
Order_id infos	+ Add	Change
Patient_addrs	+ Add	Change
Patient_extra datas	+ Add	Change
Patient_guardians	+ Add	Change
Patients	+ Add	Change
Pregnancys	+ Add	Change
Prescriptions	+ Add	Change
Problems	+ Add	Change
Provenances	+ Add	Change
Provider Phoness	+ Add	Change
Provider_id infos	+ Add	Change
Providers	+ Add	Change
Social historys	+ Add	Change

- Click on a row to view the detailed information for the record that was loaded.

<input type="checkbox"/>	Natural key	Name
<input type="checkbox"/>	ORDER-PROVIDER-1	Stephen, Dr Morrissey M
<input type="checkbox"/>	FAKE-88499	,
<input type="checkbox"/>	FAKE-13537	Mayer, Tim
<input type="checkbox"/>	FAKE-97455	Gordon, Raphael
<input type="checkbox"/>	FAKE-82795	Roberts, Michael
<input type="checkbox"/>	FAKE-63426	Sally, Michael
<input type="checkbox"/>	FAKE-13371	Mayer, Ross
<input type="checkbox"/>	FAKE-68968	McVetta, Jane
<input type="checkbox"/>	FAKE-94708	Klompas, Mike
<input type="checkbox"/>	FAKE-77023	Lazarus, Jason

5. Review the fields that were populated (and not populated).

Provenance:	epicpro.esp.08102017 2017-08-11 11:23:19.941986 ubuntu1404  
Natural key:	<input type="text" value="ORDER-PROVIDER-1"/> <small>Unique Record identifier in source EMR system</small>
Last Name:	<input type="text" value="Stephen"/>
First Name:	<input type="text" value="Morrissey"/>
Middle Name:	<input type="text" value="M"/>
Title:	<input type="text" value="Dr"/>
Primary Department identifier in source EMR system:	<input type="text" value="Lab1"/>
Primary Department:	<input type="text" value="Fenway Health"/>
Primary Department Address 1:	<input type="text" value="1340 Bolyston Street"/>
Primary Department Address 2:	<input type="text"/>
Primary Department City:	<input type="text" value="Boston"/>
Primary Department State:	<input type="text" value="MA"/>
Primary Department Zip:	<input type="text" value="02215"/>
Primary Department	<input type="text" value="617"/>

6. Items to Review:

- Are all expected fields populated?
- Is data appearing the correct field?
- Is the data formatted correctly?
- Any items missing?

6 Using the Database to Validate: Member

Login to the database using psql or a utility that gives you sql command line access and run the following commands:

1. Count of patients

Query:

```
select count(*) from emr_patient;
```

Questions/Comments:

Does the count match the expected number?

2. Check for null MRN

Query:

```
select count(*) from emr_patient where mrn is null or mrn in ('', '');
```

Questions/Comments:

Do an exceptionally large number of patients not have an MRN?

3. Check for null Last Name

Query:

```
select count(*) from emr_patient where last_name is null or last_name in ('', '');
```

Questions/Comments:

Are there any patients without a last name?

4. Verify gender counts and values

Query:

```
select count(*), gender
from emr_patient
group by gender;
```

Questions/Comments:

Gender should typically be populated for all patients and values should be one of (M, F, T, or U)

5. Verify race counts and values

Query:

```
select count(*), race
from emr_patient
group by race;
```

Questions/Comments:

Race should typically be populated for all patients and values should be one of the following:

```
CAUCASIAN
W
WHITE
BLACK
B
OTHER
O
MULTIRACIAL
```

INDIAN
I
AMERICAN INDIAN/ALASKAN NATIVE
ASIAN
A
NAT AMERICAN
NATIVE HAWAI
PACIFIC ISLANDER/HAWAIIAN
P
ALASKAN
HISPANIC

7 Using the Database to Validate: Provider

Login to the database using psql or a utility that gives you sql command line access and run the following commands:

1. Count of providers

Query:

```
select count(*) from emr_provider;
```

Questions/Comments:

Does the count match the expected number?

2. Count of patients assigned as a PCP for each provider

Query:

```
select count(patient.id) patient_count, pro.id provider_id  
from emr_provider pro, emr_patient patient  
where pro.id = patient.pcp_id  
group by pro.id;
```

Questions/Comments:

Are patients matching up to providers?

Are a large number of patients assigned to pcp_id of 1 (unknown)?

8 Using the Database to Validate: Visits

Login to the database using psql or a utility that gives you sql command line access and run the following commands:

1. Count of visits

Query:

```
select count(*) from emr_encounter;
```

Questions/Comments:

Does the count match the expected number?

2. Count of encounters per diagnosis code

Query:

```
select count(*), dx_code_id from emr_encounter e, emr_encounter_dx_codes dx
```

```
where e.id = dx.encounter_id
group by dx_code_id;
```

Questions/Comments:

Are diagnosis codes being included and correctly parsed for encounters?

3. Count of records with pregnancy information

Query:

```
select count(*), pregnant
from emr_encounter
group by pregnant;
```

Questions/Comments:

Is pregnancy information being populated interpreted based on extract data?

4. Count of records with EDD (estimated due date) populated.

Query:

```
select count(*),
case when edd is not null then 'edd_populated' else null end as
edd_check
from emr_encounter
group by edd_check;
```

If EDD is provided, does it match the general number of encounters expected?

5. Count of records and values with height and weight and bmi

Queries:

```
select count(*), raw_height
from emr_encounter
group by raw_height;
```

```
select count(*), raw_weight
from emr_encounter
group by raw_weight;
```

```
select count(*), raw_bmi
from emr_encounter
group by raw_bmi;
```

Questions/Comments:

Are fields being populated? Do the formats match the spec? Note, some sites send raw_bmi as opposed to height and weight so not having these values may not be of concern.

9 Using the Database to Validate: Medications

Login to the database using psql or a utility that gives you sql command line access and run the following commands:

1. Count of medications

Query:

```
select count(*) from emr_prescription;
```

Questions/Comments:

Does the count match the expected number?

2. Count of records without a start date

Query:

```
select count(*) from emr_prescription where start_date is null;
```

Questions/Comments:

Start Date is important so it really should be populated for true rx records. If the count of null records is high, please investigate.

3. Count of records without a start date

Query:

```
select count(*) from emr_prescription where end_date is null;
```

Questions/Comments:

End Date is often not populated, but ideally it should be sent as it is important for computing rx spans.

10 Using the Database to Validate: Lab Results

Login to the database using psql or a utility that gives you sql command line access and run the following commands:

1. Count of lab results

Query:

```
select count(*) from emr_labresult;
```

Questions/Comments:

Does the count match the expected number?

2. Count of lab tests by native_name

Query:

```
select count(*), native_name, procedure_name
from emr_labresult
group by native_name, procedure_name;
```

Questions/Comments:

Do the native names appear as expected? Do the number of tests appear reasonable per test?

Note: this may not be advised on systems with a large amount of lab results

3. Count of records with specimen source populated

Query:

```
select count(*), specimen_source from emr_labresult
group by specimen_source;
```

Questions/Comments:

At least some tests should have a specimen source.

4. Count of records without a result string

Query:

```
select count(*) from emr_labresult
where result_string is null or result_string in ('', '');
```

Questions/Comments:

Ensure that most lab results have a result value.

11 Using the Database to Validation: Historical Data Load

Login to the database using psql or a utility that gives you sql command line access and run the following commands:

1. Count of lab result records with breakdown by month and year

Query:

```
select count(*), date_part('year', date) as dateyear, date_part('month',
date) as datemon
from emr_labresult
group by dateyear, datemon order by dateyear, datemon;
```

2. Count of medication records with breakdown by month and year

Query:

```
select count(*), date_part('year', date) as dateyear, date_part('month',
date) as datemon
from emr_prescription
group by dateyear, datemon order by dateyear, datemon;
```

3. Count of encounter records with breakdown by month and year

Query:

```
select count(*), date_part('year', date) as dateyear, date_part('month',
date) as datemon
from emr_encounter
group by dateyear, datemon order by dateyear, datemon;
```

12 Troubleshooting Load Failures

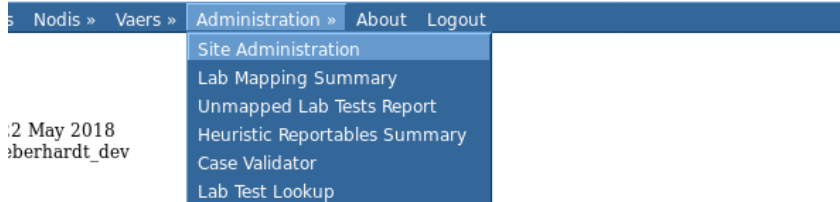
1. Carefully review any error messages displayed in the screen or in the log file (var/log/esp.log). Often the error message will provide detailed information to identify the issue.
2. For errors relating to a field exceeding the length of the column, this can sometimes happen if data file contains a header row or the fields are not populated in the correct order. Check the file to ensure that
 - a) it does not contain header row and
 - b) the data fields line up with the FileSpec document.If a field length does need to be expanded to support correct EMR data, please contact CII for assistance.
3. If no error message is displayed, however the load report is reporting 0 rows were loaded, ensure that a file with this filename has not been loaded before. Each loaded file name must be unique and only loaded once in ESP. To “reload” a file that has previously been loaded in ESP, use the following command:

```
/srv/esp/prod/bin load_epic --reload
```

13 Deleting Test Data

1. To fully delete the test file data from ESP, login to the ESP UI.
2. Navigate to “Administration” -> “Site Administration”

ESP — ELECTRONIC MEDICAL RECORD SUPPORT



3. In the “EMR” section, select “Provenances”

Emr	
Allergys	+ Add Change
Encounter type maps	+ Add Change
Encounters	+ Add Change
Hospital_problems	+ Add Change
Immunizations	+ Add Change
Lab Orders	+ Add Change
Lab Specimen Observations	+ Add Change
Lab Specimens	+ Add Change
Lab Test Results	+ Add Change
Lab infos	+ Add Change
Order Extensions	+ Add Change
Order_id infos	+ Add Change
Patient_addr	+ Add Change
Patient_extra datas	+ Add Change
Patient_guardians	+ Add Change
Patients	+ Add Change
Pregnancys	+ Add Change
Prescriptions	+ Add Change
Problems	+ Add Change
Provenances	+ Add Change
Provider Phonest	+ Add Change
Provider_id infos	+ Add Change
Providers	+ Add Change
Social historys	+ Add Change

4. The following are special provenance files for ESP use. **DO NOT DELETE** the following:

<input type="checkbox"/>	2	Jan. 1, 1900, midnight	CLEANUP	(None)
<input type="checkbox"/>	1	Jan. 1, 1900, midnight	SYSTEM	(None)

5. Place a checkbox next to the data file you want to delete and then choose “Delete Selected Provenances” from the dropdown menu. **Reminder: DO NOT DELETE Provenances 1 and 2.**

Select provenance to change

Action: ----- 1 of 96 selected

<input type="checkbox"/>	Prov	Timestamp
<input checked="" type="checkbox"/>	393	Aug. 11, 2017, 11:23 a.m.
<input type="checkbox"/>	392	Aug. 11, 2017, 11:23 a.m.
<input type="checkbox"/>	391	Aug. 11, 2017, 11:23 a.m.
<input type="checkbox"/>	390	Aug. 11, 2017, 11:23 a.m.

6. Click “Go”
7. Review the data to confirm what will be deleted. Please note, deleting one type of record may delete additional data that may be tied to record being deleted. If satisfied, select “Yes I’m sure”.
8. Please note, large files may take a few minutes to display and/or delete.