

# PDO Query Use Case Scenarios

## Scenario: Get patient data from a patient set id

This request is divided into three parts:

1. input\_list

The input\_list accepts either the id of the patient set or a list of patient ids.

1. filter\_list

The filter\_list holds a list of panels. The panels have the item details that are used in constructing a PDO query.

1. output\_option

The output\_option specifies which set of patient data to return.

Each of the patient data sections in the *output\_option* has attributes to specify the level of detail data that is expected in the response.

Request Type	Request	Response
getPDO_fromInputList	GetPDOFromInputList_requestType	master_responseType

## Input Option List Type

The input list to the PDO request can be in one of the following formats. i.e. the patient\_list, encounter\_list, pid\_list, and eid\_list.

Element Name	Description
patient_list	<p>The patient_list input can be in one of these three forms.</p> <ol style="list-style-type: none"><li>1. The patient set id (&lt;patient_set_coll_id&gt;)</li><li>2. An enumeration list of the patient id (&lt;patient_id&gt;)</li><li>3. The entire patients of datamart (&lt;entire_patient_set&gt;)</li></ol> <p><b>Example for &lt;patient_list&gt;:</b></p> <pre>&lt;patient_list max="10" min="0"&gt; &lt;patient_id index="0"&gt;86850&lt;/patient_id&gt;  &lt;!--OR--&gt;  &lt;patient_set_coll_id&gt;4741&lt;/patient_set_coll_id&gt;  &lt;!--OR--&gt;  &lt;entire_patient_set &gt;true&lt;/entire_patient_set&gt; &lt;/patient_list&gt;</pre>
encounter_list	<p>The encounter_list input format is similar to the patient_list. This list can be in one of three forms.</p> <ol style="list-style-type: none"><li>1. The encounter set id (&lt;encounter_set_coll_id&gt;)</li><li>2. An enumeration list of the encounter id (&lt;encounter_id&gt;)</li><li>3. The entire encounters of datamart (&lt;entire_encounter_set&gt;)</li></ol>
pid_list	<p>The pid_list consists of a list of enumerated patient ids (patient_id) and their source value. This input allows the user to query the data via the patient_id and its source. The PDO service first maps the patient_id to the patient_num before querying the datamart.</p> <p><b>Example for &lt;pid_list&gt;:</b></p> <pre>&lt;pid_list&gt; &lt;pid index="1" source="MGH_E"&gt;PAT_MGH_001&lt;/pid&gt; &lt;pid index="2" source="BWH_E"&gt;PAT_BWH_001&lt;/pid&gt; &lt;/pid_list&gt;</pre>

eid_list	<p>The eid_list consists of a list of enumerated encounter ids (encounter_ide) and their source value. This input allows the user to query the data via the encounter_ide and its source. The PDO service first maps the encounter_ide to the encounter_num before querying the datamart.</p> <p><b>Example for &lt;pid_list&gt;:</b></p> <pre>&lt;eid_list&gt; &lt;eid index="1" source="MGH_E"&gt;ENC_MGH_001&lt;/pid&gt; &lt;eid index="2" source="BWH_E"&gt;ENC_BWH_001&lt;/pid&gt; &lt;/eid_list&gt;</pre>
----------	---

## Filter List Type

Please refer to the earlier section titled *Panel Definition Detail*.

The LARGETEXT search in the panel definition requires the minimum of DATA\_DEID user role.

## Output Option List Type

This option specifies the list of sections that is required in the PDO response.

Attribute Name	Description	Implemented	
name = "asattributes"	none	This option specifies to return the names for the codes when returning the PDO data. The names for the codes are available in the CODE_LOOKUP table.	Yes
phi = "encrypted"	unencrypted		No
time = "nozone"	withzone		No

### Example:

```
<!-- output options -->
<output_option name="asattributes">
<patient_set select="using_filter_list" onlykeys="true"/>
<concept_set_using_filter_list select="using_filter_list" onlykeys="true"/>
<modifier_set_using_filter_list select="using_filter_list" onlykeys="true"/>
<observation_set selectionfilter="min_value" withmodifiers="true" blob="false" onlykeys="false"/>
<event_set select="using_filter_list" onlykeys="true"/>
<pid_set select="using_filter_list" onlykeys="true"/>
<eid_set select="using_filter_list" onlykeys="true"/>
<observer_set_using_filter_list onlykeys="true"/>
</output_option>
```

The following are the available options for each element in the *<output\_option>*.

Attribute Name	Description	Implemented
onlykeys = "true"	false	If this option is "true" then only the primary keys will be returned. If it is "false" then all the fields except the blob and tech data fields will be returned.
blob = "true"	false	This option specifies whether the blob field is required in the output. The blob option requires the minimum of DATA_DEID user role.
techdata = "true"	false	This option specifies whether the blob field is required in the output. The blob option requires the minimum of DATA_DEID user role.
select = "using_filter_list"	using_input_list	There are a couple of ways to select the PDO data.  If the select option is "using_input_list" then the returned data will be based on the <input_list> and the fact table will not be used to return the data.  If it is "using_filter_list" then return the data that satisfies the <filter_list> constraints when applied on the fact table.

selectionfilter = "min_value"	max_value	first_value	last_value	single_observation	last_n_values	<p>This option will retrieve specific results depending on which one is defined in the xml message.</p> <p>If <i>min_value</i> is specified then the minimum for the NVAL_NUM will be returned for each patient in the concept group.</p> <p>If <i>max_value</i> is specified then the maximum for the NVAL_NUM will be returned for each patient in the concept group.</p> <p>If <i>first_value</i> is specified then the minimum for the START_DATE will be returned for each patient in the concept group.</p> <p>If <i>last_value</i> is specified then the maximum for the START_DATE will be returned for each patient in the concept group.</p> <p>If <i>single_observation</i> is specified then the will be return just one record even if it contains more than one due to different values in the MODIFIER_CD column for each patient in the concept group.</p> <p>If <i>last_n_values</i> is specified then the last number of records in the concept group, where the value contains the number of records</p>
-------------------------------	-----------	-------------	------------	--------------------	---------------	---

Element Name	Description		
patient_set	Return the set of patient dimension data either for the input list or from the patient present in the observation set.		
observation_set	Return the observation set of the patient data object. There could be a multiple number of <i>&lt;observation_set&gt;</i> returned and the number of <i>&lt;observation_set&gt;</i> returned will be equal to the number of panels defined in the filter list.  Observation set has the attribute "panel_name" which corresponds to "name" attribute defined in the <i>&lt;panel&gt;</i> .  For backward compatibility to 1.5 clients, which expects a unique observation fact not including the modifier option of the fact namely the INSTANCE_NUM column, the sending application version should have 1.5 in the PDO request.		
Attribute	Description		
withmodifiers = "true"	false"		If the value is "false" then the service will return the observation where the modifier_cd = '@'. The default value is "true", returns the observation irrespective of the modifier_cd value.
ue	Return the set containing event / visit dimension data occurring in the observation set or from the input list.		
pid_set	Return the patient mapping table's information for the given input list or from the observation set.		
patient_id	patient_id_source	patient_num	patient_id_status
1000000	HIVE	1000000	A
1000001	HIVE	1000001	A
123	MGH	1000001	A
777	BWH	1000001	A

```

<pid_set>
<!--pid with only the hive number -->
<pid>
<patient_id status="A" source="HIVE">1000000</patient_id>
</pid>
<!--pidwithhivenumberandmappingpatient_id's-->
<pid>
<patient_id status="A" source="HIVE">1000001</patient_id>
<patient_map_id status="A" source="MGH">123</patient_map_id>
<patient_map_id status="A" source="BWH">777</patient_map_id>
</pid>
</pid_set>

```

eid_set	Return the encounter mapping table's information for the given input list or from the observation set.
observer_set_using_filter_list	Return the set containing observer / provider dimension data occurring in the observation set.
concept_set_using_filter_list	Return the set of concept dimension data occurring in the observation set.

modifier_set_using_filter_list	Return the set of modifier dimension data occurring in the observation set.
--------------------------------	---

**Example:**

```
<message_body>
<crc.pdoheader>
<request_type>getPDO_fromInputList</request_type>
</crc.pdoheader>
<crc:request xsi:type="ns2:GetPDOFromInputList_requestType"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<!-- inputlist -->
<input_list>
<patient_list max="300" min="1">
<patient_set_coll_id>100</patient_set_coll_id>
</patient_list>
<!--or-->
<event_list/>
<!--or-->
<pid_list min="1" max="3">
<pid index="1" source="HIVE">9876</pid>
<pid index="3" source="HIVE">1</pid>
</pid_list>
<!--or-->
<eid_list min="1" max="5">
<eid index="1" source="HIVE">12222313</eid>
</eid_list>
</input_list>
<!-- filterlist -->
<filter_list>
<panel name="panel1">
<panel_invert>0</panel_invert>
<panel_accuracy_scale></panel_accuracy_scale>
<panel_start_date></panel_start_date>
<panel_end_date></panel_end_date>
<item>
<item_name></item_name>
<item_key></item_key>
<constrain_by_value>
<value_operator></value_operator>
<value_constraint>value_constraint0</value_constraint>
<value_unit_of_measure>unit</value_unit_of_measure>
<value_type></value_type>
</constrain_by_value>
<constrain_by_modifier>
<applied_path>\i2b2\Medications%</applied_path>
<modifier_key>\i2b2_DEMO\Dose{color:#0000ff}</modifier_key>
<constrain_by_value>
<value_operator></value_operator>
<value_constraint> </value_constraint>
<value_unit_of_measure></value_unit_of_measure>
<value_type></value_type>
</constrain_by_value>
</constrain_by_modifier>
<constrain_by_date>
</constrain_by_date>
</item>
...
</panel>
<panel name="panel2">
<panel_invert>0</panel_invert>
<panel_accuracy_scale></panel_accuracy_scale>
<panel_start_date></panel_start_date>
<panel_end_date></panel_end_date>
<item>
<item_name> </item_name>
<item_key> </item_key>
<constrain_by_value>
<value_type></value_type>
<value_operator></value_operator>
<value_unitofmeasure></value_unitofmeasure>
<value></value>
</constrain_by_value>
<constrain_by_modifier></constrain_by_modifier>
<constrain_by_date></constrain_by_date>
</item>
</panel>
...
</filter_list>
```

```

<!-- output options -->
<output_option>
<patient_set select="using_filter_list" onlykeys="true"/>
<concept_set_using_filter_list select="using_filter_list" onlykeys="true"/>
<modifier_set_using_filter_list select="using_filter_list" onlykeys="true"/>
<observation_set blob="false" onlykeys="false"/>
<event_set select="using_filter_list" onlykeys="true"/>
<pid_set select="using_filter_list" onlykeys="true"/>
<eid_set select="using_filter_list" onlykeys="true"/>
<observer_set_using_filter_list onlykeys="true"/>
<!-- To specify generalized dimension type -->
<dimension_set_using_filter_list dimensionname="dimension1" onlykeys="true"/>
</output_option>
</crc:request>
<!-- response begin -->
<response>
<patient_data>
<!-- patient set section begins -->
<patient_set>
<patient>
<patient_id>patient_id6</patient_id>
<param column="vital_status_cd" type="string"
column_descriptor="date interpretation code"
name="">param3</param>
<param column="birth_date" type="dateTime"
column_descriptor="birthdate"
name="">param3</param>
<param column="death_date" type="dateTime"
column_descriptor="date of death">param3</param>
<param column="sex_cd" type="string"
column_descriptor="gender" name="">param3</param>
<param column="age_in_years_num" type="int"
column_descriptor="age">param3</param>
<param column="language_cd" type="string"
column_descriptor="language" name="">param3</param>
<param column="race_cd" type="string"
column_descriptor="race" name="">param3</param>
<param column="religion_cd" type="string"
column_descriptor="religion" name="">param3</param>
<param column="marital_status_cd" type="string"
column_descriptor="marital status" name="">param3</param>
<param column="statecityzip_path_char" type="string"
column_descriptor="zip code hierarchy"
name="">param3</param>
</patient>
...
</patient_set>
<!-- concept set section begins -->
<concept_set>
<concept>
<concept_path>concept_path0</concept_path>
<concept_cd>concept_cd0</concept_cd>
<name_char>name_char0</name_char>
</concept>
...
</concept_set>
<!-- observation set section begins -->
<observation_set panel_name="panel1">
<observation>
<event_id source="source3">event_id3</event_id>
<patient_id>patient_id9</patient_id>
<concept_cd name="name0">concept_cd3</concept_cd>
<observer_cd source="source0">observer_cd3</observer_cd>
<start_date>2006-05-04T18:13:51.0Z</start_date>
<modifier_cd name="name1">modifier_cd0</modifier_cd>
<instance_num>1</instance_num>
<valuetype_cd>valuetype_cd0</valuetype_cd>
<tval_char>tval_char0</tval_char>
<nval_num units="units0">3.141592653589</nval_num>
<valueflag_cd name="name2">valueflag_cd0</valueflag_cd>
<quantity_num>3.141592653589</quantity_num>
<units_cd>units_cd0</units_cd>
<end_date>2006-05-04T18:13:51.0Z</end_date>
<location_cd name="name3">location_cd0</location_cd>
</observation>
</observation>
...
</observation>
</observation_set>

```

```

<observation_set panel_name="panel2">
<observation>
...
</observation>
<observation>
...
</observation>
...
</observation_set>
<!-- event set section begins -->
<event_set>
<event>
<event_id source="source0">event_id0</event_id>
<patient_id>patient_id0</patient_id>
<start_date>2006-05-04T18:13:51.0Z</start_date>
<end_date>2006-05-04T18:13:51.0Z</end_date>
<param column="inout_cd" type="string" name=""
column_descriptor="in vs. outpatient code"></param>
<param column="location_cd" type="string" name=""
column_descriptor="location code"></param>
<param column="location_path" type="string" name=""
column_descriptor="location hierarchy"></param>
<param column="active_status_cd" type="string" name=""
column_descriptor="date interpretation code"></param>
</event>
...
</event_set>
<!-- observer / provider set section begins -->
<observer_set>
<observer>
<observer_path>observer_path0</observer_path>
<observer_cd>observer_cd0</observer_cd>
<name_char>name_char3</name_char>
</observer>
...
</observer_set>
<!-- pid set section begins -->
<pid_set>
<pid>
<patient_id status="A" source="HIVE">123456</patient_id>
</pid>
<pid>
<patient_id status="A" source="HIVE">234567</patient_id>
<patient_map_id status="A" source="EMP_E">ABC1</patient_map_id>
<patient_map_id status="A" source="MGH_E">XYZ1</patient_map_id>
</pid>
...
</pid_set>
</patient_data>
</response>
<!-- response end -->
</message_body>

```

Scenario: Get observation blob by primary key

This request returns the observation blob using the observation primary key.

**Example:**

```
<message_body>
<pdoheader>
<request_type>get_observationfact_by_primary_key</request_type>
</pdoheader>
<ns5:request xsi:type="ns5:GetObservationFactByPrimaryKey_requestType" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<fact_primary_key>
<event_id>2004005981</event_id>
<patient_id>52003</patient_id>
<concept_cd>LCS-I2B2:c1009c</concept_cd>
<observer_id>03840261</observer_id>
<start_date>1995-08-24T00:00:00.179-05:00</start_date>
<instance_num>1</instance_num>
</fact_primary_key>
<fact_output_option select="using_filter_list" onlykeys="false"/>
</ns5:request>
<response>
<patient_data>
<observation_set>
<observation>
<event_id source="source3">event_id3</event_id>
<patient_id>patient_id9</patient_id>
<concept_cd name="name0">concept_cd3</concept_cd>
<observer_cd source="source0">observer_cd3</observer_cd>
<start_date>2006-05-04T18:13:51.0Z</start_date>
<instance_num>1</instance_num>
<observation_blob>

<![CDATA[
patient notes]]>
</observation_blob>
</observation >
</observation_set>
</patient_data>
</response>
</message_body>
```