BACKGROUND

This website contains publicly available trial documents from the Patent Trial and Appeal Board (PTAB).

PTAB Bulk Data Services is a RESTful API that will provide a client user interface for browsing the USPTO Patent Trial and Appeal Board data, formulating queries, and requesting one-time bulk downloads of publically available trial documents. Furthermore, it will allow developers to take advantage of custom search syntax to develop their own applications.

WHO IS PTAB BULK DATA SERVICES FOR?

PTAB Bulk Data Services is designed for developers and technical users worldwide to publish their inventions for international use and adoption. It is open to third party developers inside and outside of the government, so that they can directly benefit from this data, by making and using their own apps. There is no registration process or API keys required to use this service.

Overview

This search engine is intended to be used by anyone interested in reviewing decisions rendered by PTAB The following shows advanced search features designed to help in analyzing decision data.

Terms

A query is broken up into terms and operators. There are two types of terms: Single Terms and Phrases.

A Single Term is a single word such as "CFR" or "Patent".

A Phrase is a group of words surrounded by double quotes such as "Art Unit 3669".

Multiple terms can be combined together with Boolean operators to form a more complex query (see below).

Fields

USPTO Decision text search supports fielded data. When performing a search you can either specify a field, or use the default field. The field names and default field is implementation specific.

You can search any field by typing the field name followed by a colon ":" and then the term you are looking for.

As an example, let's assume a USPTO Decision Search index contains two fields, title and text and text is the default field. If you want to find the document entitled "The Right Way" which contains the text "don't go this way", you can enter:

```
applicationNumber:13845920 AND ocrText:AIA
```

or

```
applicationNumber:"13845920" AND AIA
```

Since text is the default field, the field indicator is not required.

Note: The field is only valid for the term that it directly precedes, so the query

```
ocrText:AIA first invent
```

Will only find "AIA" in the ocrText field. It will find "first" and "invent" in the default field (in this case the text field).

Please use below JSON Name to perform a search on specific fields.

Field Name/Facet	JSON Name
Proceeding #	proceedingNumber
PO/Respondent	respondentPartyName (or) partyName
Board Decision	subdecisionTypeCategory
Decision Date	documentFilingDate
PO/Respondent Application #	respondentApplicationNumberText (or) applicationNumberText
Proceeding Type	proceedingTypeCategory
Decision type	decisionTypeCategory
PO/Respondent Art Unit	respondentGroupArtUnitNumber (or) groupArtUnitNumber
Accessibility	proceedingConfidentiality
PO/Respondent patent #	respondentPatentNumber (or) patentNumber
PO/Respondent patent grant date	respondentGrantDate (or) grantDate

PO/Respondent patent owner name	respondentPatentOwnerName (or) patentOwnerName
PO/Respondent patent inventor name	respondentInventorName (or) inventorName
PO/Respondent party name	respondentPartyName (or) partyName
PO/Respondent counsel name	respondentCounselName (or) counselName
Petitioner application #	petitionerApplicationNumberText (or) applicationNumberText
Petitioner patent #	petitionerPatentNumber (or) patentNumber
Petitioner patent grant date	petitionerGrantDate (or) grantDate
Petitioner patent owner name	petitionerPatentOwnerName (or) patentOwnerName
Petitioner patent inventor name	petitionerInventorName (or) inventorName
Petitioner party name	petitionerPartyName (or) partyName
Petitioner counsel name	petitionerCounselName (or) counselName
Petitioner art unit	petitionerGroupArtUnitNumber (or) groupArtUnitNumber
Proceeding Status	proceedingStatusCategory
Proceeding Filing Date	proceedingFilingDate

Accorded filing date	accordedFilingDate
Institution decision date	institutionDecisionDate
Last modified date	proceedingLastModifiedDate
Document type	documentTypeName
Title	documentTitleText
Document Filing Date	documentFilingDate
Filing party	filingPartyCategory
Document/exhibit #	documentNumber
Document size	documentSize
Media type	mediaTypeCategory
PO/Respondent Tech Center	respondentTechnologyCenterNumber (or) technologyCenterNumber
Petitioner Tech Center	petitionerTechnologyCenterNumber (or) technologyCenterNumber
Sub Proceeding Type	subproceedingTypeCategory
Issue Type	issueType
Board Rulings	boardRulings

Term Modifiers

USPTO Decision search supports modifying query terms to provide a wide range of searching options.

Wildcard Searches

USPTO Office Action search supports single and multiple character wildcard searches within single terms (not within phrase queries).

To perform a single character wildcard search use the "?" symbol.

To perform a multiple character wildcard search use the "*" symbol.

The single character wildcard search looks for terms that match that with the single character replaced. For example, to search for "text" or "AIA" you can use the search:

```
A?A
```

Multiple character wildcard searches looks for 0 or more characters. For example, to search for AIA, AbA you can use the search:

```
A*A
```

You can also use the wildcard searches in the middle of a term.

```
te*t
```

Note: You cannot use a * or? Symbol as the first character of a search.

Fuzzy Searches

USPTO Decision Search supports fuzzy searches based on the Levenshtein Distance, or Edit Distance algorithm. To do a fuzzy search use the tilde, "~", symbol at the end of a Single word Term. For example to search for a term similar in spelling to "patent" use the fuzzy search:

```
patent~
```

This search will find terms like patenting, patented, patents, etc.

Starting with USPTO Decision Search 1.9 an additional (optional) parameter can specify the required similarity. The value is between 0 and 1, with a value closer to 1 only terms with a higher similarity will be matched. For example:

```
patent~0.8
```

The default that is used if the parameter is not given is 0.5.

Proximity Searches

USPTO Decision Search supports finding words are a within a specific distance away. To do a proximity search use the tilde, "~", symbol at the end of a Phrase. For example to search for a "Art" and "Unit" within 3 words of each other in a document use the search:

```
"Art Unit"~3
```

Range Searches

Range Queries allow one to match documents whose field(s) values are between the lower and upper bound specified by the Range Query. Range Queries can be inclusive or exclusive of the upper and lower bounds. Sorting is done lexicographically.

```
accordedFilingDate:[2002-01-01 TO 2003-01-01]
```

This will find documents whose mailing date fields have values between 2002-01-01 and 2003-01-01, inclusive. Note that Range Queries are not reserved for date fields. You could also use range queries with non-date fields.

Boosting a Term

USPTO Decision Search provides the relevance level of matching documents based on the terms found. To boost a term use the caret, "^", symbol with a boost factor (a number) at the end of the term you are searching. The higher the boost factor, the more relevant the term will be.

Boosting allows you to control the relevance of a document by boosting its term. For example, if you are searching for

```
Art Unit
```

and you want the term "Art" to be more relevant boost it using the ^ symbol along with the boost factor next to the term. You would type:

```
Art^4 Unit
```

This will make documents with the term Art appear more relevant. You can also boost Phrase Terms as in the example:

```
"Art Unit"^4 "3669"
```

By default, the boost factor is 1. Although the boost factor must be positive, it can be less than 1 (e.g. 0.2)

Boolean Operators

Boolean operators allow terms to be combined through logic operators. USPTO Decision Search supports AND, OR, NOT and "-" as Boolean operators (Note: Boolean operators must be ALL CAPS).

The OR operator is the default conjunction operator. This means that if there is no Boolean operator between two terms, the OR operator is used. The OR operator links two terms and finds a matching document if either of the terms exist in a document. This is equivalent to a union using sets. The symbol || can be used in place of the word OR.

To search for documents that contain either "Art Unit" or just "Art" use the query:

```
"Art Unit" Art
```

or

```
"Art Unit" OR Art
```

or

```
"Art Unit" || Art
```

AND

The AND operator matches documents where both terms exist anywhere in the text of a single document. This is equivalent to an intersection using sets. The symbol && can be used in place of the word AND.

To search for documents that contain "Art Unit" and "3669" use the query:

```
"Art Unit" AND "3669"
```

or

```
"Art Unit" && "3669"
```

NOT

The NOT operator excludes documents that contain the term after NOT. This is equivalent to a difference using sets.

To search for documents that contain "Art Unit" but not "3669" use the query:

```
"Art Unit" NOT "3669"
```

-

The "-" or prohibit operator excludes documents that contain the term after the "-" symbol.

To search for documents that contain "Art Unit" but not "3669" use the query:

```
"Art Unit" -"3669"
```

Grouping

USPTO Office Action Search supports using parentheses to group clauses to form sub queries. This can be very useful if you want to control the BOOLEAN logic for a query.

To search for either "Art" or "Unit" and "website" use the query:

```
(Art OR Unit) AND website
```

This eliminates any confusion and makes sure you that website must exist and either term Art or Unit may exist.

Field Grouping

USPTO Decision Search supports using parentheses to group multiple clauses to a single field.

To search for a ocrText that contains both the word "double" and the phrase "35 U.S.C.101" use the query:

```
ocrText:(+double +"35 U.S.C.101")
```

Escaping Special Characters

USPTO Decision Search supports escaping special characters that are part of the query syntax. The current list special characters are

+ - && || ! ( ) [ ] ^ " ~ * ? : \

To escape these character use the \ before the character. For example to search for (1+1):2 use the query:

```
\(1\+1\)\:2
```