

Cnibr 检索字段及运算符

一、 检索字段

Cnibr 中文平台可检索 40 多个字段，具体的字段如下表。

| 字段名 | 别名 |
|----------|---------------|
| 申请号 | an |
| 申请日 | ad |
| 公开（公告）号 | pnm |
| 公开（公告）日 | pd |
| 专利号 | pn |
| 专利类型 | pat |
| 名称 | ti |
| 摘要 | abst |
| 权利要求书 | clm |
| 主权项 | cl |
| 说明书 | de |
| 分类号 | sic |
| 主分类号 | pic |
| 欧洲分类号 | sec |
| 欧洲主分类号 | pec |
| 范畴分类 | |
| 申请（专利权）人 | pa |
| 发明（设计）人 | inn |
| 专利代理机构 | agc |
| 代理人 | agt |
| 地址 | ar |
| 国省代码 | co |
| 申请来源 | apf |
| 国际申请 | ian |
| 国际公布 | ipn |
| 进入国家日期 | den |
| 法律状态 | law |
| 专利权状态 | patstatus |
| 专利权状态代码 | patstatuscode |
| 优先权 | pri |
| 优先权号 | prn |
| 优先权日 | prd |
| 同族专利项 | fa |
| 参考文献 | refrens |
| 分案原申请号 | dan |
| 审查员 | examiner |

| | |
|--------|-----------|
| 颁证日 | issuedate |
| 公开年 | pdy |
| 最新法律状态 | |

二、 检索运算符

运算符是表达式的不可缺少的组成部分,负责将多个检索目标按照一定的逻辑连接起来。

Cnipro的检索运算符包括以下几种。

(1) 比较运算符

| 符号 | 含义和作用 | 举例 |
|----|-------|-------------|
| = | 等于 | 申请人=华为 |
| != | 不等于 | 申请人!=华为 |
| > | 大于 | 申请日 > 2000 |
| < | 小于 | 申请日 < 2000 |
| >= | 大于等于 | 申请日 >= 2000 |
| <= | 小于等于 | 申请日 <= 2000 |

- 比较运算符作用于cnipro检索平台的所有字段。
- 比较运算符的左值必须是一个的检索入口（字段名或其别名）。
- 如果检索条件不是单个词，而是一个表达式，则该表达式应该用圆括号“(”和“)”括起来。

(2) 逻辑运算符

| 符号 | 含义和作用 | 举例 |
|-----|-----------|-----------|
| and | 两者必须同时满足 | 中国 and 美国 |
| xor | 两者只能满足其一 | 中国 xor 美国 |
| not | 两者中只能出现前者 | 中国 not 美国 |
| or | 两者至少满足其一 | 中国 or 美国 |

- 逻辑运算符不仅能够连接两个检索词，也可以连接两个检索子表达式，从而构造出非常复杂的检索表达式。

(3) 重复逻辑运算符

| 符号 | 含义和作用 | 举例 |
|----|------------------|----------------------|
| *= | 检索各个字段上满足同一条件的记录 | c1,c2 *= (中国 and 香港) |
| ^= | 检索只有一个字段上满足条件的记录 | c1,c2 ^= (中国 and 香港) |
| += | 检索任意字段上同时满足条件的记录 | c1,c2 += (中国 and 香港) |

- 重复逻辑运算符实际上是在各个指定的字段上用相同的条件式进行检索后,再在各个字段之间进行指定的逻辑运算。
- 重复逻辑运算符的左值必须是由单字节的逗号 ‘,’ 分隔的两个或两个以上的检索入口 (字段名或其别名)。如果右值不是单个词,而是一个表达式 (该运算需要在这个表达式上进行), 则该表达式应该用圆括号 “(” 和 “)” 括起来。

(4) 属性运算符

| 符号 | 含义和作用 | 举例 |
|-----------|--------------------|-----------------|
| adj | 两者邻接先后同时出现 | 控制 adj 设备 |
| equ/n | 两者正好相距n个位置先后同时出现 | 控制 equ/5 设备 |
| pre/n | 两者相距至多n个位置先后同时出现 | 控制 pre/5 设备 |
| pre/n# | 两者正好相距n个位置先后同时出现 | 控制 pre/5# 设备 |
| pre/pos=n | | 控制 pre/pos=5 设备 |
| pre/nL | 两者相距至少n个位置先后同时出现 | 控制 pre/5L 设备 |
| pre/sen | 两者在同一句中先后同时出现 | 控制 pre/sen 设备 |
| pre/seg | 两者在同一段中先后同时出现 | 控制 pre/seg 设备 |
| and/n | 两者相距至多n个位置同时出现 | 控制 and/5 设备 |
| and/n# | 两者正好相距n个位置同时出现 | 控制 and/5# 设备 |
| and/pos=n | | 控制 and/pos=5 设备 |
| and/nL | 两者相距至少n个位置同时出现 | 控制 and/5L 设备 |
| and/sen | 在同一句中两者同时出现 | 控制 and/sen 设备 |
| and/seg | 在同一段中两者同时出现 | 控制 and/seg 设备 |
| xor/n | 同一句中不能在n个位置之内同时出现 | 控制 xor/2 设备 |
| xor/n# | 同一句中不能正好相距n个位置同时出现 | 控制 xor/2 设备 |
| xor/pos=n | | 控制 xor/pos=2 设备 |
| xor/nL | 同一句中不能在n个位置之外同时出现 | 控制 xor/2L 设备 |
| xor/sen | 在同一句中只能出现其一 | 控制 xor/sen 设备 |
| xor/seg | 在同一段中只能出现其一 | 控制 xor/seg 设备 |

- 在属性运算符equ/n、X/n、X/n#、X/nL和X/pos=n中的距离是指两个字/词之间所间隔的“位置”的个数,而不是其首“位置”的差值。
- 属性运算符X/sen=n和X/seg=n中的n, 都从0开始计数。
- 属性运算符adj等价于pre/0; equ/n等价于pre/n#。

(5) 运算符的等价符号

用英文单词书写运算符，有时觉得比较麻烦，可用下列等价符号进行替代：

| | | | | | | | |
|-------|----|-----|-----|-----|-----|-----|----|
| 常规运算符 | != | ADJ | PRE | AND | XOR | NOT | OR |
| 等价运算符 | <> | 空格 | & | * | ^ | - | + |

(6) 通配符

| 符号 | 含义和作用 | 举例 |
|----|---------------|--------|
| ? | 一个字母字符或汉字 | 作者=张?庆 |
| % | 表示0到多个字母字符或汉字 | 作者=张% |

通配符用于模糊搜索文件，cnipr 的通配符包含“?”和“%”。