T-constraint review

T-constraints are a Backstage-developed searching tool to help make quality control just a little easier. With t-constraints we can look for existence of data, omission of data, or combinations of some data existing with other data missing. The possibilities are endless, but we have just a small little cheat-sheet available to help you construct some basic search commands.

t-constraint key

t=	Search for following tag
[2]t=	At least two of this tag exist (you can put any numerical digit in the brackets)
s=	Search for following subfield
[2]s=	At least two of this subfield exist within the tag
i=	Search for the following indicators, bracket the digits
=1	Previous information exists
!=1	Previous information does not exist
=3	Search for the following Case Sensitive information
!=3	The following information does not exist (Case sensitive)
=4	Search for the following non-case sensitive information
!=4	Following phrase does not exist (not case sensitive)
[a-z]	Looking for uncapped letters from a-z
[A-Z]	Looking for Capped letters from A-Z
[0-9]	Looking for numbers 0-9
?	Variable information (typically used with indicator information)
^	Tag or subfield begins with the following information (must be placed at the beginning of your “specific information” phrase
$	Tag or subfield ends with the previous information (must be placed at the end of your “specific information” phrase)
\	Includes the following punctuation
[15,3]	15 is position in 008. 3 is length
&	Include both items in your search
|	Either one OR the other included in your search, but not necessarily both

1. Always start with largest piece of information (i.e., the tag number to search, t=) and work your way down to most specific. Tag numbers can be variable such as XXX to search all tags, 6XX to search any 6xx, or X10, etc.

2. Text information (words, punctuation, etc.) needs to go in quotation marks (" "). This information must be preceded by !=3, =3, !=4, or =4

3. Each t-constraint MUST contain at least t= or [2]t=

T-constraint review

Order of information/Examples: *<u>Always</u> start with largest piece of information first and work your way down to the most specific*.

TAG, EXIST?

`t=546 =1`

searches for records that include a 546 tag

`t=6XX !=1`

searches for records that do not include a 6xx tag

`[2]t=949 =1`

searches for records with at least 2 949 tags

TAG, SUBFIELD, EXIST?

`t=700 s=e =1`

searches for a 700 with a $e

`t=260 s=b !=1`

searches for 260 tags that are missing $b

TAG, SUBFIELD, EQUALS?, "SPECIFIC INFORMATION"

`t=949 =3 "REFERENCE"`

searches the 949 tags for the case-sensitive phrase "REFERENCE"

`t=050 s=ab =3 "^[a-z]" | t=050 s=ab =3 "^\.[a-z]"`

*searches for 050 tags that have EITHER a subfield **a** or **b** that begins with a lowercase letter OR a period and lowercase letter*

`t=245 s=c =4 "trans"`

searches for "trans" anywhere in the 245c (will find trans., translator, translation, etc.)

`t=773 =4 "\.$"`

searches for a 773 tag that ends with a period

`t=ldr =4 "m"`

searches for an "m" in the leader, not case sensitive, so this would pick up both the BibLvl and ELvl

TAG, 008 POSITION, "SPECIFIC INFORMATION"

`t=008 [15,1] =4 "x" & t=26x s=a =1 & t=26x s=a !=4 "identified|S\.l\."`

searches for the 008 country code that starts with "x" but there is a 26X $a that exists that does not have the phrase "Place of publication not identified" or the Latin abbreviation "S.l."

***Note, that when you want to include punctuation as part of your search string, you will need to precede it with \ to indicate we're looking for the actual character and not a variable.*

TAG, INDICATORS, EXIST?

`t=650 i=[?5] =1`

searches for 650 tags that have second indicator 5

`t=264 i=[?4] & [2]t=264 !=1`

searches for a 264 with 2nd indicator 4 where there is not a 2nd 264 present in the record.

The question mark denotes that any value can be in the first indicator position

T-constraint review

Practice: Create a t-constraint based on the following scenarios (use the previous examples and table for hints)…

1. I want to look for a serial record.

2. I need to find records that don’t have a $v in the 490

3. I want to find 300 tags that have no pagination.

4. I need to look to see if I put “cases” in a $v in my 650 tags

5. I need to find records that have a 546 but don’t have an 041

6. I need to find 501 with notes.

7. Do my original cataloging records need a gov. code in the 008 for my 110 tags?

8. My 541 tags cannot end with a period, I need to look for that.

9. BONUS: I have a medium of performance listed in my 240 field but I want to find those with a 382 but no subfield s in that field

T-constraint review

Practice ANSWERS:

1. I want to look for a serial record.
   t=ldr =3 "s"

2. I need to find records that don’t have a $v in the 490
   t=490 s=v !=1

3. I want to find 300 tags that have no pagination.
   t=300 s=a =4 "^p\. :" | t=300 s=a !=1

4. I need to look to see if I put “cases” in a $v in my 650 tags
   t=600 s=v =3 "cases"

5. I need to find records that have a 546 but don’t have an 041
   t=546 =1 & t=041 !=1

6. I need to find 501 with notes.
   t=501 =1

7. Do my records need a gov. code in the 008 for my 110 tags?
   t=110 i=[1?] = 1& t=008 [28,1] =4 " "

8. My 541 tags cannot end with a period, I need to look for that.
   t=541 =4 "\.$"

9. BONUS: I have a medium of performance listed in my 240 field but I want to find those with a 382 but no subfield s in that field
   t=240 s=m =1 & t=382 =1 & t=382 s=s !=1