

## 5.60 cardinality\_atmost\_partition

	DESCRIPTION	LINKS	GRAPH
Origin	Derived from <a href="#">global_cardinality</a> .		
Constraint	<code>cardinality_atmost_partition(ATMOST, VARIABLES, PARTITIONS)</code>		
Type	VALUES : <code>collection(val-int)</code>		
Arguments	ATMOST : <code>dvar</code> VARIABLES : <code>collection(var-dvar)</code> PARTITIONS : <code>collection(p - VALUES)</code>		
Restrictions	$ VALUES  \geq 1$ <a href="#">required</a> (VALUES, val) <a href="#">distinct</a> (VALUES, val) $ATMOST \geq 0$ $ATMOST \leq  VARIABLES $ <a href="#">required</a> (VARIABLES, var) <a href="#">required</a> (PARTITIONS, p) $ PARTITIONS  \geq 2$		
Purpose	ATMOST is the maximum number of time that values of a same partition of PARTITIONS are taken by the variables of the collection VARIABLES.		
Example	$(2, \langle 2, 3, 7, 1, 6, 0 \rangle, \langle p - \langle 1, 3 \rangle, p - \langle 4 \rangle, p - \langle 2, 6 \rangle \rangle)$ <p>In this example, two variables of the collection <math>VARIABLES = \langle 2, 3, 7, 1, 6, 0 \rangle</math> are assigned values of the first partition, no variable is assigned a value of the second partition, and finally two variables are assigned values of the last partition. As a consequence, the <code>cardinality_atmost_partition</code> constraint holds since its first argument ATMOST is assigned to the maximum number of occurrences 2.</p>		
Typical	$ATMOST > 0$ $ATMOST <  VARIABLES $ $ VARIABLES  > 1$ $ VARIABLES  >  PARTITIONS $		
Symmetries	<ul style="list-style-type: none"> <li>Items of VARIABLES are <a href="#">permutable</a>.</li> <li>Items of PARTITIONS are <a href="#">permutable</a>.</li> <li>Items of PARTITIONS.p are <a href="#">permutable</a>.</li> </ul>		
Arg. properties	Functional dependency: ATMOST determined by VARIABLES and PARTITIONS.		

**See also**

**generalisation:** `global_cardinality` (*single count variable replaced by an individual count variable for each value and variable replaced by variable  $\in$  partition*).

**used in graph description:** `in`.

**Keywords**

**characteristic of a constraint:** partition.

**constraint arguments:** pure functional dependency.

**constraint type:** value constraint.

**filtering:** arc-consistency.

**final graph structure:** acyclic, bipartite, no loop.

**modelling:** at most, functional dependency.

Arc input(s)	VARIABLES PARTITIONS
Arc generator	<i>PRODUCT</i> $\mapsto$ collection(variables, partitions)
Arc arity	2
Arc constraint(s)	in(variables.var, partitions.p)
Graph property(ies)	<i>MAX_ID</i> = ATMOST
Graph class	<ul style="list-style-type: none"><li>• ACYCLIC</li><li>• BIPARTITE</li><li>• NO_LOOP</li></ul>

**Graph model** Parts (A) and (B) of Figure 5.152 respectively show the initial and final graph associated with the **Example** slot. Since we use the *MAX\_ID* graph property, a vertex with the maximum number of predecessor is stressed with a double circle.

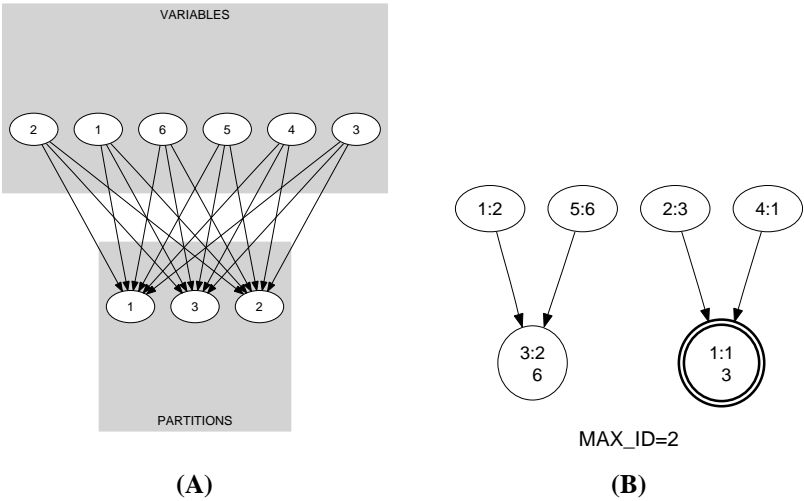


Figure 5.152: Initial and final graph of the cardinality\_atmost\_partition constraint

20030820

771