5.415 used_by_partition

DESCRIPTION LINKS GRAPH

Origin Derived from used_by.

Constraint used_by_partition(VARIABLES1, VARIABLES2, PARTITIONS)

Type
```
VALUES : collection(val−int)
```

Arguments
```
VARIABLES1 : collection(var−dvar)
VARIABLES2 : collection(var−dvar)
PARTITIONS : collection(p − VALUES)
```

Restrictions
```
|VALUES| ≥ 1
required(VALUES, val)
distinct(VALUES, val)
|VARIABLES1| ≥ |VARIABLES2|
required(VARIABLES1, var)
required(VARIABLES2, var)
required(PARTITIONS, p)
|PARTITIONS| ≥ 2
```

Purpose For each integer i in $[1, |\texttt{PARTITIONS}|]$, let $N1_i$ (respectively $N2_i$) denote the number of variables of VARIABLES1 (respectively VARIABLES2) that take their value in the i^{th} partition of the collection PARTITIONS. For all i in $[1, |\texttt{PARTITIONS}|]$ we have $N2_i > 0 \Rightarrow N1_i \geq N2_i$.

Example
$$\left(\begin{array}{l}\langle 1,9,1,6,2,3\rangle,\\ \langle 1,3,6,6\rangle,\\ \langle \mathtt{p}-\langle 1,3\rangle, \mathtt{p}-\langle 4\rangle, \mathtt{p}-\langle 2,6\rangle\rangle\end{array}\right)$$

The different values of the collection VARIABLES2 $= \langle 1,3,6,6\rangle$ are respectively associated with the partitions $\mathtt{p}-\langle 1,3\rangle$, $\mathtt{p}-\langle 1,3\rangle$, $\mathtt{p}-\langle 2,6\rangle$, and $\mathtt{p}-\langle 2,6\rangle$. Therefore partitions $\mathtt{p}-\langle 1,3\rangle$ and $\mathtt{p}-\langle 2,6\rangle$ are respectively used 2 and 2 times.

Similarly, the different values of the collection VARIABLES1 $= \langle 1,9,1,6,2,3\rangle$ (except value 9, which does not occur in any partition) are respectively associated with the partitions $\mathtt{p}-\langle 1,3\rangle$, $\mathtt{p}-\langle 1,3\rangle$, $\mathtt{p}-\langle 2,6\rangle$, $\mathtt{p}-\langle 2,6\rangle$, and $\mathtt{p}-\langle 1,3\rangle$. Therefore partitions $\mathtt{p}-\langle 1,3\rangle$ and $\mathtt{p}-\langle 2,6\rangle$ are respectively used 3 and 2 times.

Consequently, the used_by_partition constraint holds since, for each partition associated with the collection VARIABLES2 $= \langle 1,3,6,6\rangle$, its number of occurrences within VARIABLES1 $= \langle 1,9,1,6,2,3\rangle$ is greater than or equal to its number of occurrences within VARIABLES2:

- Partition $\mathtt{p}-\langle 1,3\rangle$ occurs 3 times within $\langle 1,9,1,6,2,3\rangle$ and 2 times within $\langle 1,3,6,6\rangle$.

- Partition p $-$ $\langle 2, 6 \rangle$ occurs 2 times within $\langle 1, 9, 1, 6, 2, 3 \rangle$ and 2 times within $\langle 1, 3, 6, 6 \rangle$.

Typical

$|\texttt{VARIABLES1}| > 1$
$\texttt{range}(\texttt{VARIABLES1.var}) > 1$
$|\texttt{VARIABLES2}| > 1$
$\texttt{range}(\texttt{VARIABLES2.var}) > 1$
$|\texttt{VARIABLES1}| > |\texttt{PARTITIONS}|$
$|\texttt{VARIABLES2}| > |\texttt{PARTITIONS}|$

Symmetries

- Items of VARIABLES1 are permutable.
- Items of VARIABLES2 are permutable.
- Items of PARTITIONS are permutable.
- Items of PARTITIONS.p are permutable.
- An occurrence of a value of VARIABLES1.var can be replaced by any other value that also belongs to the same partition of PARTITIONS.
- An occurrence of a value of VARIABLES2.var can be replaced by any other value that also belongs to the same partition of PARTITIONS.

Arg. properties

- Contractible wrt. VARIABLES2.
- Extensible wrt. VARIABLES1.
- Aggregate: VARIABLES1(union), VARIABLES2(union), PARTITIONS(id).

Used in k_used_by_partition.

See also **implied by:** same_partition.
soft variant: soft_used_by_partition_var *(variable-based violation measure)*.
specialisation: used_by *(*variable ∈ partition *replaced by* variable*)*.
system of constraints: k_used_by_partition.
used in graph description: in_same_partition.

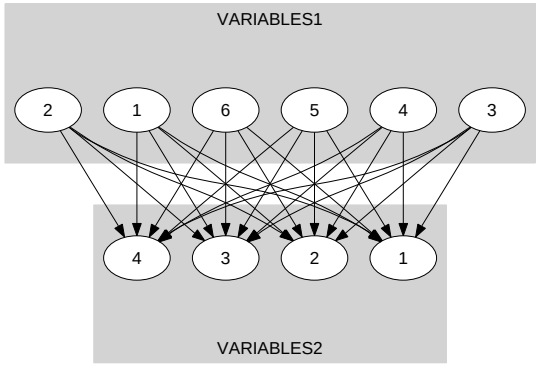
Keywords **characteristic of a constraint:** partition, sort based reformulation.
constraint arguments: constraint between two collections of variables.
modelling: inclusion.

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	*PRODUCT* $\mapsto$ collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	in_same_partition(variables1.var, variables2.var, PARTITIONS)
Graph property(ies)	• for all connected components: **NSOURCE**$\geq$**NSINK** • **NSINK**$=$ |VARIABLES2|

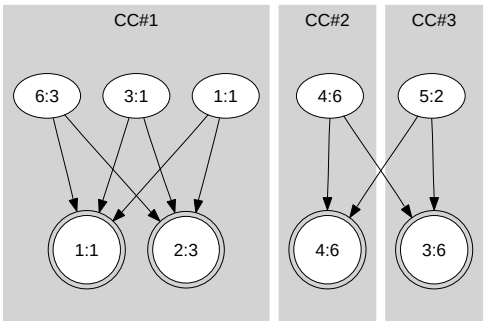
Graph model Parts (A) and (B) of Figure 5.787 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NSOURCE** and **NSINK** graph properties, the source and sink vertices of the final graph are stressed with a double circle. Since there is a constraint on each connected component of the final graph we also show the different connected components. Each of them corresponds to an equivalence class according to the arc constraint. Note that the vertex corresponding to the variable that takes value 9 was removed from the final graph since there is no arc for which the associated equivalence constraint holds. The used_by_partition constraint holds since:

- For each connected component of the final graph the number of sources is greater than or equal to the number of sinks.
- The number of sinks of the final graph is equal to |VARIABLES2|.

Signature Since the initial graph contains only sources and sinks, and since sources of the initial graph cannot become sinks of the final graph, we have that the maximum number of sinks of the final graph is equal to |VARIABLES2|. Therefore we can rewrite $\mathbf{NSINK} = |\text{VARIABLES2}|$ to $\mathbf{NSINK} \geq |\text{VARIABLES2}|$ and simplify $\overline{\mathbf{NSINK}}$ to $\mathbf{NSINK}$.



(A)



(B)

Figure 5.787: Initial and final graph of the `used_by_partition` constraint