# Source and Target Reusability

June 3, 2014

Table 1: Source and Target Reusability

| MappingPrimitive | Source Reusability | Target Reusability |
|---|---|---|
| ADD | N | minRequiredNumAttrs = numAddAttr + ((sk == SkolemKind.KEY) ? 1 : 0) + 1 AND keySize in the target relation should not be greater than size of relation - numAddAttr |
| ADL | minRequiredNumAttrs = keySize + numDelAttr | minRequiredNumAttrs = keySize + numAddAttr |
| CP | N | N |
| DL | minRequiredNumAttrs = keySize + numDelAttr | minRequiredNumAttrs - keySize |
| HP | minRequiredNumAttrs = 2 AND if the selected relation has primary key, it should be on the first attribute | minRequiredNumAttrs = 2 AND if the selected relation has primary key, it should be on the first attribute. Here, numberOfTargetRels = joinSize |
| ME | minRequiredNumAttrs = getNumJoinAttrs(rels.size()) + 1 AND for the first relation, the primary key should be on the NumOfJoinAttributes last attributes; for the other relations except for the last one, the primary key should be on last attributes - 2 * numOfJoinAttributes; refer to the attached example file | ((r.sizeOfAttrArray() + numTJoinAttrs) != (numOfTables * numOfAttributes[0])) AND the primary key has to be on the numTJoinAttrs last attributes. **Note:** this is a very restricted version of target reusability in ME |
| MA | same as ME | ((r.sizeOfAttrArray() - numNewAttr + numTJoinAttrs) != (numOfTables * numOfAttributes[0]) AND the primary key has to be on joinAttPos; refer to line 226 in MergeAddScenarioGenerator.java. **Note:** this is a very restricted version of target reusability in MA |
| OF | - | - [2] |
| SJ | minRequiredNumAttrs = keySize + keySize +1 AND self-referring attribute(s) should have the same type(s) | minRequiredNumAttrs1 = keySize + 1 (for S; refer to iBench First Cut) AND minRequiredNumAttrs2 = keySize + keySize (for T; refer to iBench First Cut) |

Table 2: Source and Target Reusability

| MappingPrimitive | Source Reusability | Target Reusability |
|---|---|---|
| SU | minRequiredNumAttrs = 1 AND primary key has to be on first attributes | minRequiredNumAttrs = 3 **Note:** SU does not generate target primary key |
| VP | minRequiredNumAttrs = (keySize >numOfTgtTables): keySize ? numOfTgtTables AND primary key has to be on first attributes | for the first relation: minRequiredNumAttrs = attsPerTargetRel + 1, for other relations: minRequiredNumAttrs = maxRequiredNumAttrs = attsPerTargetRel + 1 AND primary key has to be the last attribute (keySize = 1) |
| VH | minRequiredNumAttrs = numOfTgtTables | for the first relation: minRequiredNumAttrs = attsPerTargetRel + 1, for other relations: minRequiredNumAttrs = maxRequiredNumAttrs = attsPerTargetRel + 2 AND if the selected relation has primary key, size of the key should be 1 AND if it is the first relation, the primary key should be on the last attribute; otherwise, it should be on relSize - 2 |
| VI | minRequiredNumAttrs = (keySize >numOfTgtTables): keySize ? numOfTgtTables | for the first relation, minRequiredNumAttrs = (keySize ¡ attsPerTargetRel ): attsPerTargetRels ? numOfTgtTables; for the rest of the relations, minRequiredNumAttrs = maxRequiredNumAttrs = ( attsPerTargetRel + keySize ) |
| VNM | minRequiredNumAttrs = ((keySize >numOfTgtTables) : keySize ? numOfTgtTables ) | for the first relation, minRequiredNumAttrs = attsPerTargetRel + 1; for all other relations except for the last one, minRequiredNumAttrs = maxRequiredNumAttrs = attsPerTargetRel + 1; for the last one, minRequiredNumAttrs = maxRequiredNumAttrs = 2 |