abstracts more than one axiom. Thus, all of the referenced entities will be replaced by a placeholder after the application of the replacement function φ.

Figure 7: Tree showing possible variable replacements.

At this point we have demonstrated how the replacement function φ works. The next step in algorithm 1 is the computation of pairwise distances (step 5). For example, the distance \( d(B_1, B_3) = 0 \) as the application of \( \phi \) on their referencing axioms will return the same set of transformed (see Figure 8).

\[
?* \text{ SubClassOf B} \\
?owlClass \text{ SubClassOf } ?owlObjectProperty \text{ some } ?* \\
?owlClass \text{ SubClassOf } ?owlObjectProperty \text{ only } ?*
\]

Figure 8: Reason for \( d(B_1, B_3) = 0 \). Both entities have the same set of transformed axioms shown in this figure.

Finally, steps 7-13 of Algorithm 1 describe the computation of the clusters. We should comment that we selected our stopping criteria according to the maximal difference between pairs of entities. It is \( \forall \mathcal{O}, \phi, e_1, e_2 : 0 \leq d_\phi(e_1, e_2) \leq 1 \). Therefore, the algorithm will stop agglomerations when the distances between all possible pairs of elements for all clusters is equal to 1 (meaning that these entities are completely dissimilar).

**Syntactic regularities expressed with generalisations**

Finally, the description of the clusters is given by *generalisations*. Generalisations provide a synthetic view of all the axioms that contribute to generate a cluster of entities. In practice they are axioms including variables that hold similar entities. Each of these axioms can be regarded as an *instantiation* of a generalisation, as they can be obtained by replacing each variable in the generalisation with entities in the signature of...