Constraint Element Description

When you describe a constraint element the **element_name** consists of two parts separated by two hyphens.

```
part1--part2
```

Each part represents one of the algorithms the constraint is imposed on.

If the algorithm is presented as one element on a schema (like *ORF*, *Pattern*) the algorithm's part has format:

```
algorithm_element_name.unit
```

If the algorithm is presented as two subelements on a schema (like *Repeats*, *Primer*) the algorithm's part has format:

```
algorithm_element_name.left
```

or:

```
algorithm_element_name.right
```

depending on the subelement the constraint is imposed on.

Also you should specify the constraint type parameter (currently the only available type is *distance*):

```
type: distance;
```

And specify one of the distance types, for example:

```
distance-type: end-to-start;
```

Example1: The constraint is imposed on **myORF** and **myPattern** algorithm elements:

```
myORF.unit--myPattern.unit {

   type: distance;
   distance-type: start-to-start;

   # Other parameters
}
```

Example2: The constraint is imposed on **myORF** algorithm element and the left **myRepeats** algorithm subelement:

```
myORF.unit--myRepeats.left {

   type: distance;
   distance-type: start-to-end;

   # Other parameters
}
```

The available constraint elements are described in the *Constraint Elements* chapter.