Описание форматов XML-файлов с исходными и выходными данными

# Предназначение

Данный формат предназначен для описания плоской фигуры, состоящей из замкнутых контуров.

Используется как формат входных данных при работе с плоскими замкнутыми фигурами, в частности, при расчете срединной поверхности.

# Особенность геометрии

Для описания замкнутых контуров применяются кривые Безье первого, второго и третьего порядков. Кривые задаются двумя, тремя или четырьмя опорными точками соответственно порядку.

# Структура формата

Данный раздел содержит описание структуры XML-формата.

* 1. **Shape2D**

Элемент Shape2D является элементом верхнего уровня.

1. Дочерние элементы Shape2D: Contour
   * 1. **Contour**

Содержит описание замкнутого контура фигуры.

1. Элемент обязательный;
2. Максимальное количество элементов: неограниченно;
3. Родительский элемент: Shape2D;
4. Дочерние элементы:
   * JointsOfSegments
   * Segments
     + 1. **JointsOfSegments**

Агрегирует в себе набор точек стыков сегментов контура.

1. Элемент обязательный;
2. Родительский элемент: Contour;
3. Дочерние элементы: JunctionPoint.
   * + - 1. **JunctionPoint**

Задает точку стыка двух сегментов.

1. Элемент обязательный;
2. Максимальное количество элементов: неограниченно;
3. Родительский элемент: JointsOfSegments.
   * + 1. **Segments**

Агрегирует в себе набор сегментов контура фигуры.

1. Элемент обязательный;
2. Родительский элемент: Contour;
3. Дочерние элементы: BezierCurve.
   * + - 1. **BezierCurve**

Задает кривую Безье.

1. Элемент обязательный;
2. Максимальное количество элементов: неограниченно;
3. Родительский элемент: Segments;
4. Дочерние элементы: ControlPoint.

**ControlPoint**

Задает опорную точку для кривой.

1. Элемент необязательный;
2. Максимальное количество элементов: 2;
3. Родительский элемент: BezierCurve.

# Внеформатные требования

Требуется соблюдать следующие правила при описании фигуры:

* Порядок объявления точек стыков сегментов должен совпадать с порядком объявления сегментов.
* Количество точек стыков должно совпадать с количеством сегментов.
* Направление обхода контура выбирается таким образом, чтобы внутренность фигуры была справа при обходе.

# Пример

Пример описания плоской фигуры с вырезом.

<Shape2D xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="ShapeScheme.xsd">

<!--

Описание структуры основного контура фигуры

-->

<Contour>

<!-- Точки стыков сегментов -->

<Points>

<!-- Координаты точки на плоскости -->

<Point>

<X> 20 </X>

<Y> 50 </Y>

</Point>

<Point>

<X> 40 </X>

<Y> 50 </Y>

</Point>

<Point>

<X> 40 </X>

<Y> 0 </Y>

</Point>

<Point>

<X> 20 </X>

<Y> 0 </Y>

</Point>

</Points>

<!-- Описание сегментов контура -->

<Segments>

<!-- Задание кривой Безье 3-го порядка (по 4 опорным точкам) -->

<BezierCurve>

<Point>

<X> 0 </X>

<Y> 25 </Y>

</Point>

<Point>

<X> 35 </X>

<Y> 30 </Y>

</Point>

</BezierCurve>

<BezierCurve>

<Point>

<X> 35 </X>

<Y> 35 </Y>

</Point>

<Point>

<X> 32 </X>

<Y> 55 </Y>

</Point>

</BezierCurve>

<BezierCurve>

<Point>

<X> 25 </X>

<Y> 30 </Y>

</Point>

<Point>

<X> 60 </X>

<Y> 25 </Y>

</Point>

</BezierCurve>

<!-- Задание отрезка по двум точкам -->

<BezierCurve />

</Segments>

</Contour>

<!--

Описание структуры внутреннего выреза

-->

<Contour>

<Points>

<Point>

<X> 35 </X>

<Y> 15 </Y>

</Point>

</Points>

<Segments>

<BezierCurve>

<Point>

<X> 20 </X>

<Y> 20 </Y>

</Point>

<Point>

<X> 10 </X>

<Y> -5 </Y>

</Point>

</BezierCurve>

</Segments>

</Contour>

</Shape2D>