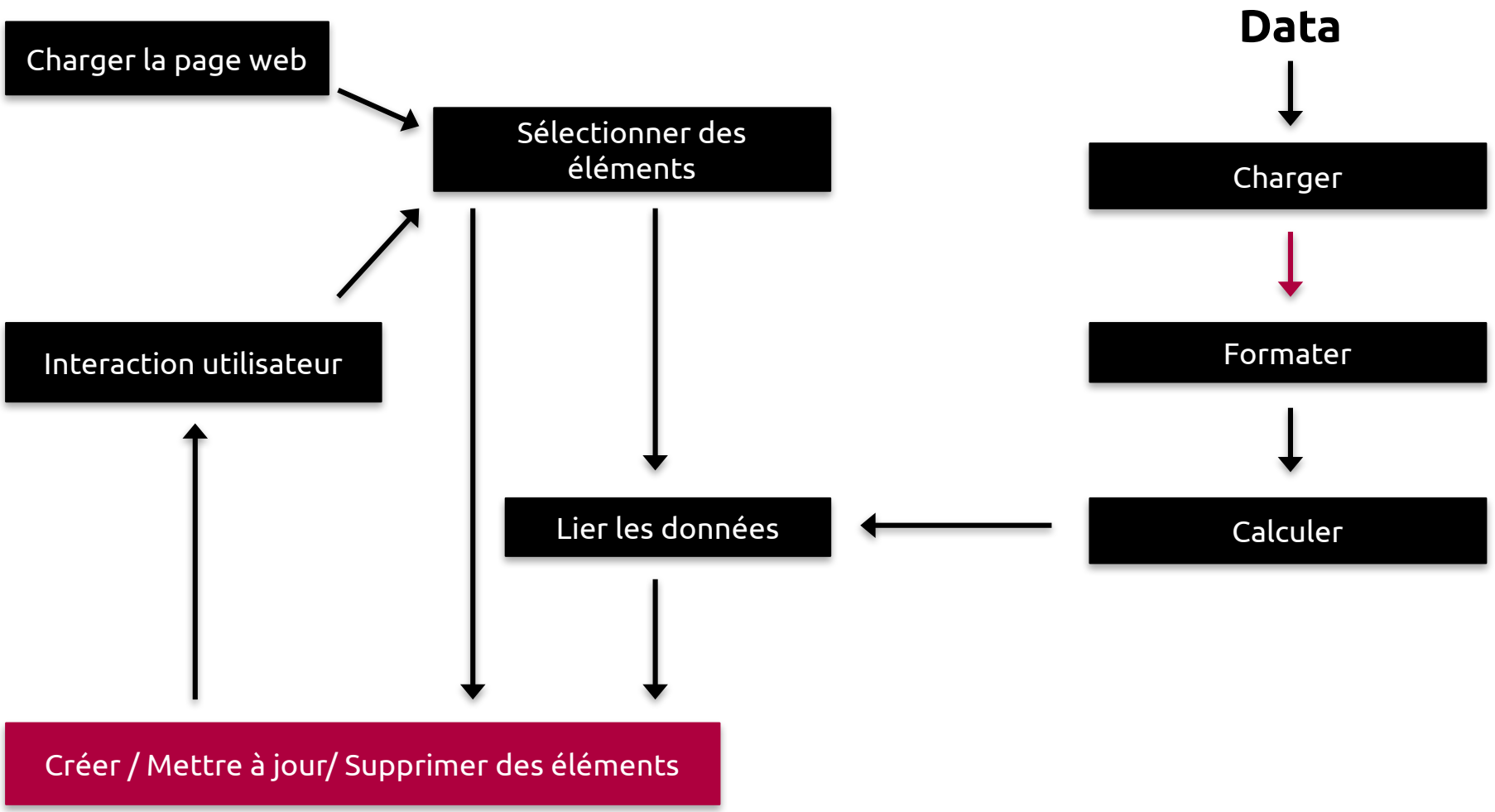
D3js

Partie 3 – d3-shape

Composants


Charger la page web
Sélectionner des éléments
Interaction utilisateur
Lier les données
Créer / Mettre à jour/ Supprimer des éléments
Data
Charger
Formater
Calculer

d3-shape

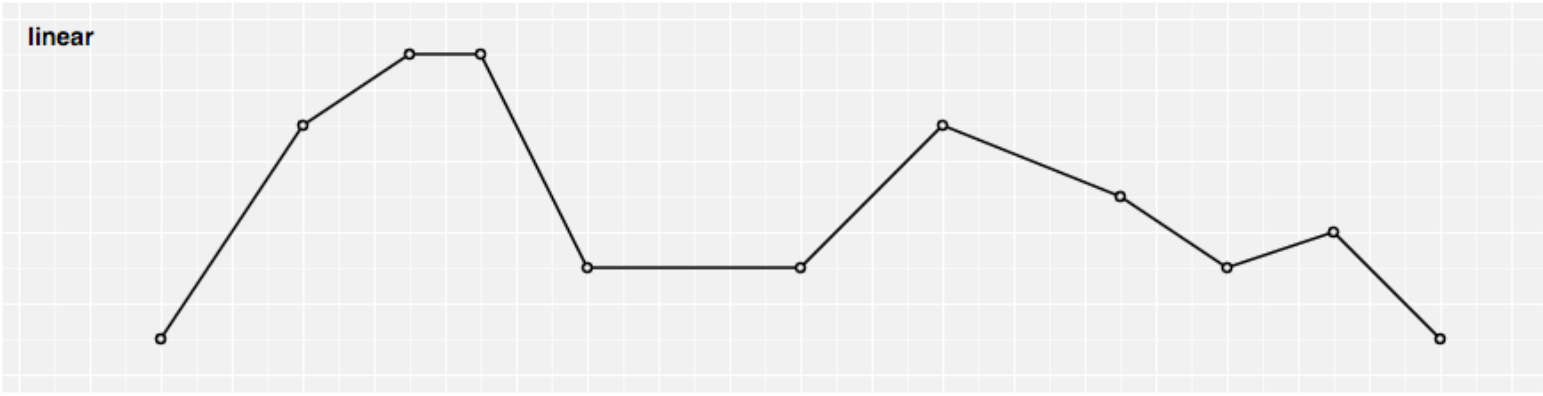
- Le projet d3-shape regroupe un ensemble de générateurs et de composants d'agencement de base.
  - **d3.line**()
  - **d3.symbol**()
  - **d3.arc**()
  - **d3.pie**()
  - **d3.area**()
  - **d3.stack**()
  - …

d3.line()

```
var myLine = d3.line()
        .x(function(d) {
                return d.jour;
        })
        .y(function(d) {
                return d.prix;
        })

d3.select("svg").append("path").attr("d", myLine(data));
```

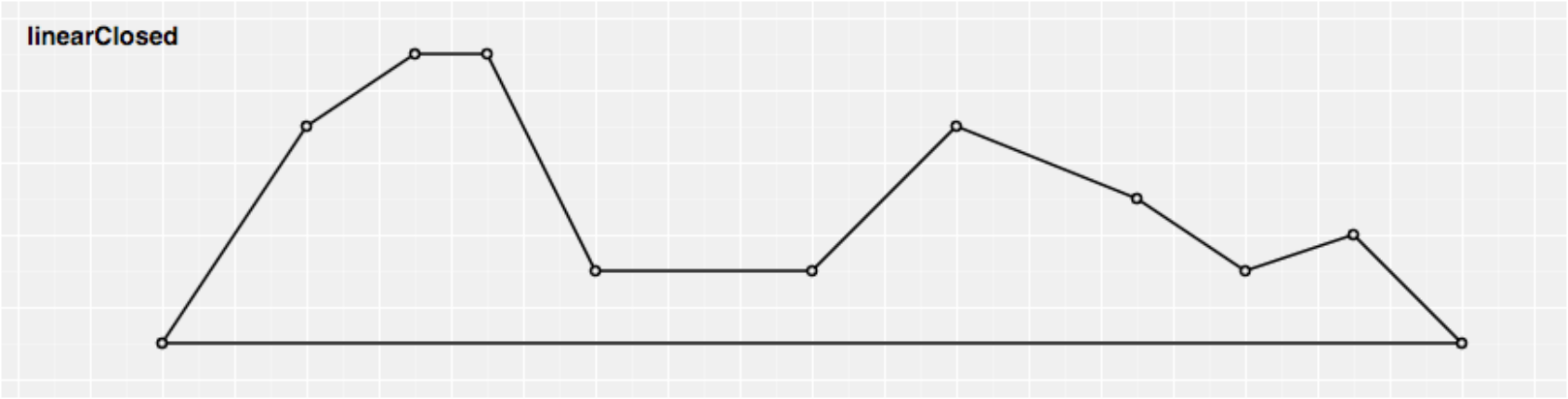
d3.curveLinear

```
var myLine = d3.line()
        .x(…)
        .y(…)
        .curve(d3.curveLinear)
…
```



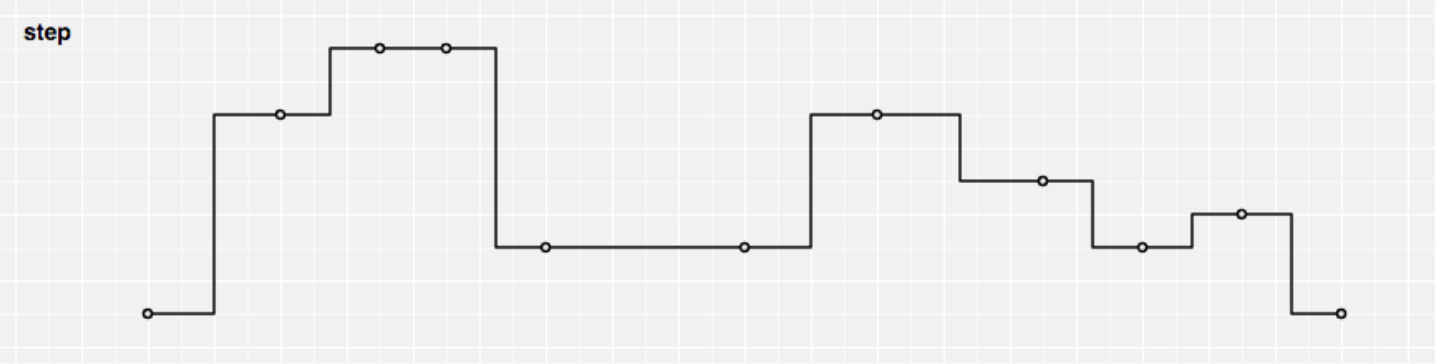
d3.curveLinearClosed

```
var myLine = d3.line()
        .x(…)
        .y(…)
        .curve(d3.curveLinearClosed)
…
```



d3.curveStep

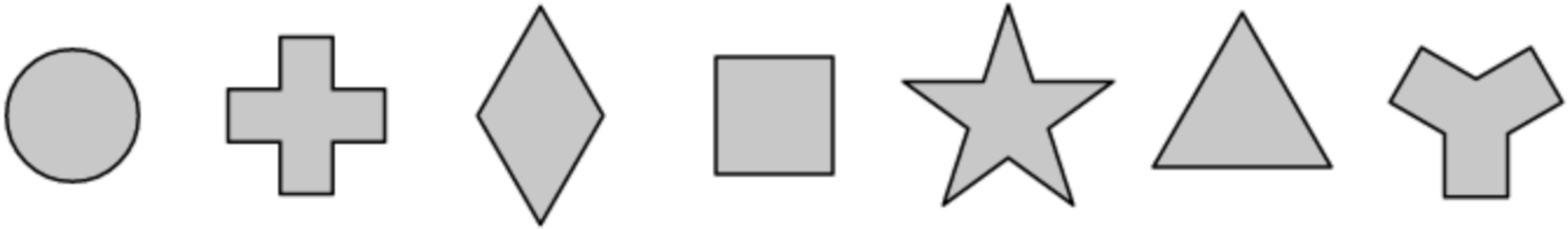
```
var myLine = d3.line()
        .x(…)
        .y(…)
        .curve(d3.curveStep)
…
```



Autres

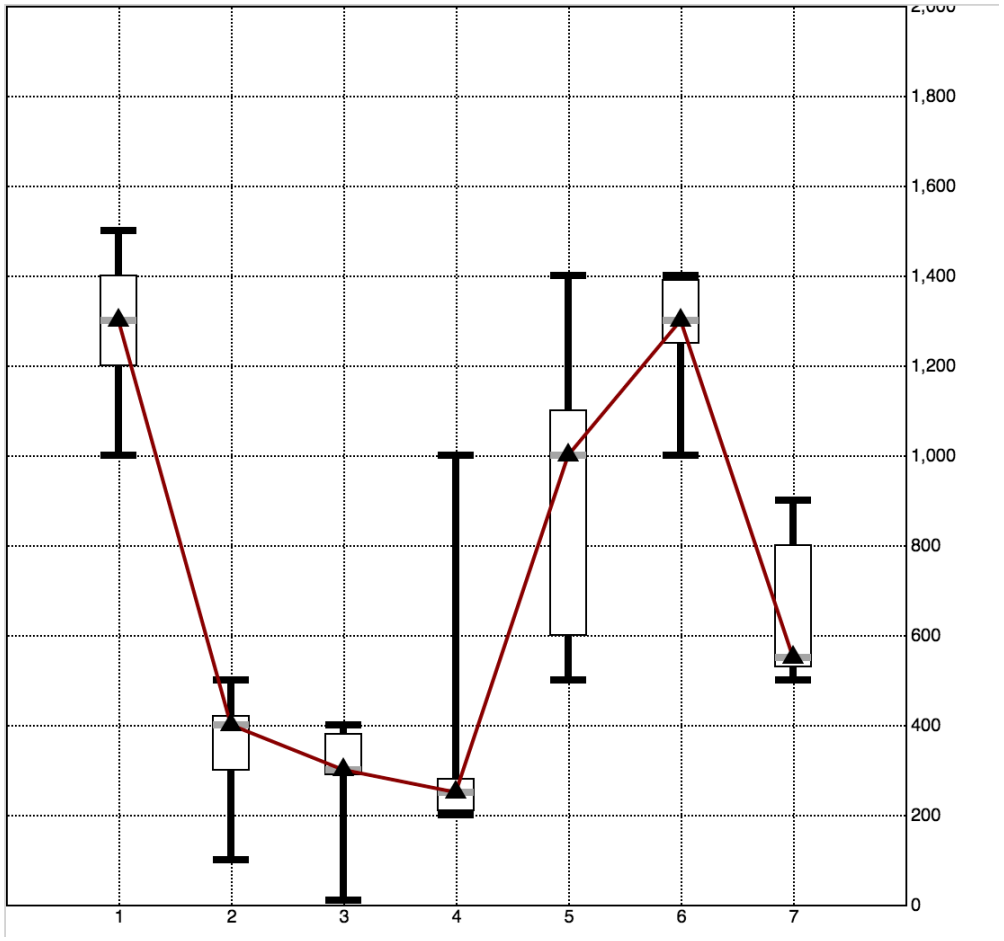
- d3.curveStepAfter()
- d3.curveStepBefore()
- d3.curveNatural()
- d3.curveMonotoneX
- d3.curveMonotoneY
- d3.curveCardinal
- Etc…

d3.symbol



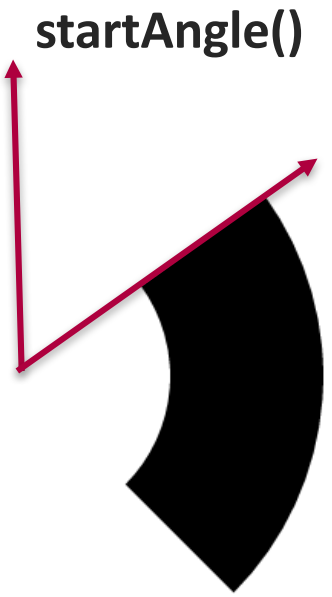
```
var symbol = d3.symbol().type(d3.symbolCircle); // d3.symbolCircle,
d3.symbolCross, d3.symbolDiamond, d3.symbolSquare, d3.symbolStar,
d3.symbolTriangle, d3.symbolWye

d3.select("svg")
  .selectAll("path.sym").data(data).enter()
      .append("path").classed("sym",true)
      .attr("d", symbol);
```

TP 7 – Les lignes



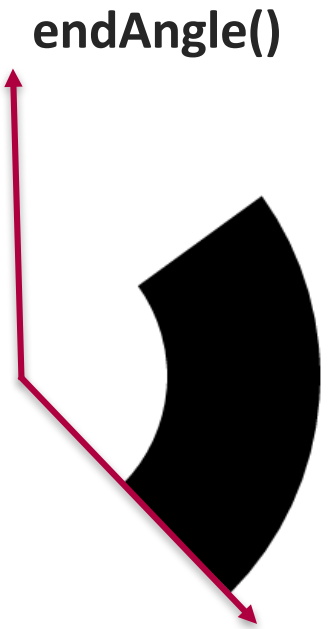
d3.arc

```
var arc = d3.arc()
            .outerRadius(100)
            .innerRadius(20)
            .startAngle(0.30*Math.PI)
            .endAngle(0.75*Math.PI);

d3.select("svg")
.append("g")
.attr("transform","translate(250,250)")
.append("path")
            .attr("d", arc);
```



d3.arc

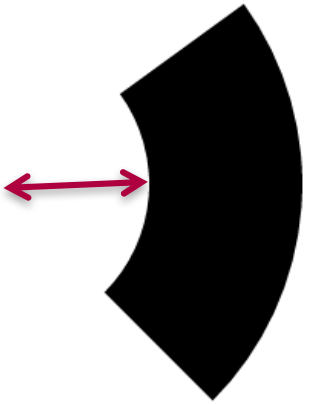
```
var arc = d3.arc()
            .outerRadius(100)
            .innerRadius(20)
            .startAngle(0.30*Math.PI)
            .endAngle(0.75*Math.PI);

d3.select("svg")
.append("g")
.attr("transform","translate(250,250)")
.append("path")
            .attr("d", arc);
```



d3.arc

```
var arc = d3.arc()
            .outerRadius(100)
            .innerRadius(20)
            .startAngle(0.30*Math.PI)
            .endAngle(0.75*Math.PI);

d3.select("svg")
.append("g")
.attr("transform","translate(250,250)")
.append("path")
            .attr("d", arc);
```

innerRadius()



d3.arc

```
var arc = d3.arc()
            .outerRadius(100)
            .innerRadius(20)
            .startAngle(0.30*Math.PI)
            .endAngle(0.75*Math.PI)
            .cornerRadius(10);

d3.select("svg")
.append("g")
.attr("transform","translate(250,250)")
.append("path")
            .attr("d", arc);
```

cornerRadius()

d3.pie

```
var pieChart = d3.pie();
var myPie = pieChart([1,1,2]);

var newArc = d3.arc().outerRadius(100).innerRadius(20);

d3.select("svg").append("g").attr("transform","translate(250,250)")
.selectAll("path").data(myPie)
                .enter()
                .append("path")
                .attr("d", newArc)
                .style("stroke", "white")
                .style("stroke-width", "3px");
```

TP 9 - Camembert

d3.area

```
var data = [ {key:10, value: 0},
{key:11, value: 20}, …]

var area = d3.area()
    .x(function(d,i) {return
xScale(d.key); })
    .y1(function(d) {
            return yScale(d.value); })
    .y0(200)

d3.select("svg").append("
path").attr("d", area(data)).style
("fill", "red");
```