1 Server-Side protocol

GSN provides several HTTP request handlers that can be used to request data about sensors. In SmartD, the main requests are received by the the /gsn (gsn.http.ControllerServlet). In SmartD several request handlers have been provided. Each of these request handlers implements the gsn.http.RequestHandler interface. Each request from the clients has a REQUEST parameter (e.g., http://server/gsn?REQUEST=type) where type is an integer. Here are the list of the parameters and the Constant representing them in SmartD.

- `118, REQUEST_SMARTD`
- `119, REQUEST_SMARTD_QUESTION`
- `120, REQUEST_SMARTD_SIMILARITY`
- `121, REQUEST_SMARTD_QUESTION_SELECTION`
- `122, REQUEST_SMARTD_NEIGHBOR_NUM_EVALUATION`
- `123, REQUEST_SMARTD_QUESTION_NUM_EVALUATION`
- `124, REQUEST_SMARTD_VALUE_DISTRIBUTION`

`REQUEST_SMARTD (http://localhost:22001/gsn?REQUEST=118&name=electric_data)`

This request is handled by the class gsn.vsensor.http.SmartDChartHandler. This servlet executes a query over the GSN's internal table and presents the results in the form of the stream elements. This request needs the following parameters:

- ids : individuals or groups of customers. The value can also be set to "all".
- startTime
- endTime
- interval: time interval

- intervalAggregation: min,max,sum or avg
- normalization: true or false
- aggregation: min,max,sum or avg

Example:

http://localhost:22001/gsn?REQUEST=118&name=electric_data&ids=1002,1018 &startTime=2009-07-19T20:00&endTime=2009-07-19T23:00 &interval=30 &intervalAggregation=max &normalization=false&aggregation=avg

Result: Figure 1.

`REQUEST_SMARTD_QUESTION (http://localhost:22001/gsn?REQUEST=119)`

This request is handled by the class gsn.vsensor.http.SmartDQuestionHandler. This servlet executes a query over the GSN's internal tables containing the questions and the answer options to closed questions. This request presents the results in the form of the stream elements. This request needs the following parameter:

- question : individual question. The value can also be set to "all".

Example:

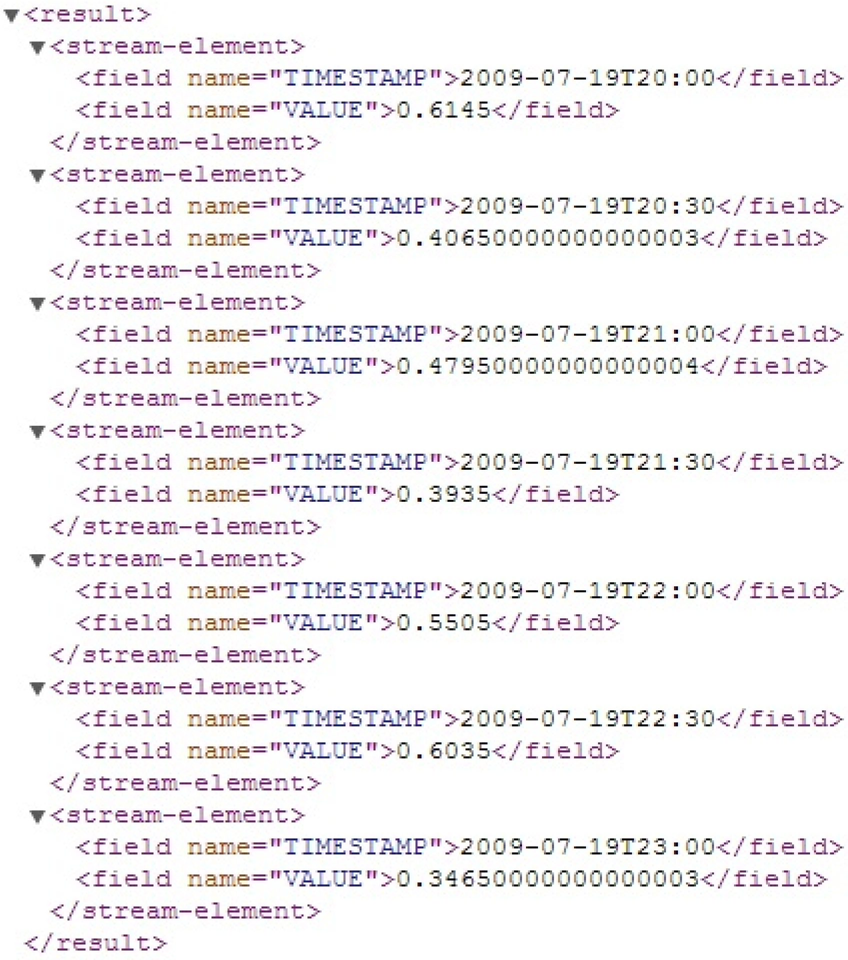
http://localhost:22001/gsn?REQUEST=119&question=8

Result: Figure 2.

`REQUEST_SMARTD_SIMILARITY (http://localhost:22001/gsn?REQUEST=120)`

This request is handled by the class gsn.vsensor.http.SmartDSimilarityHandler. This servlet executes a query over the GSN's internal table and returns the average of daily energy consumption of the similar customers in the form of the stream elements. This request needs the following parameter:

- questions : group of questions and answers.



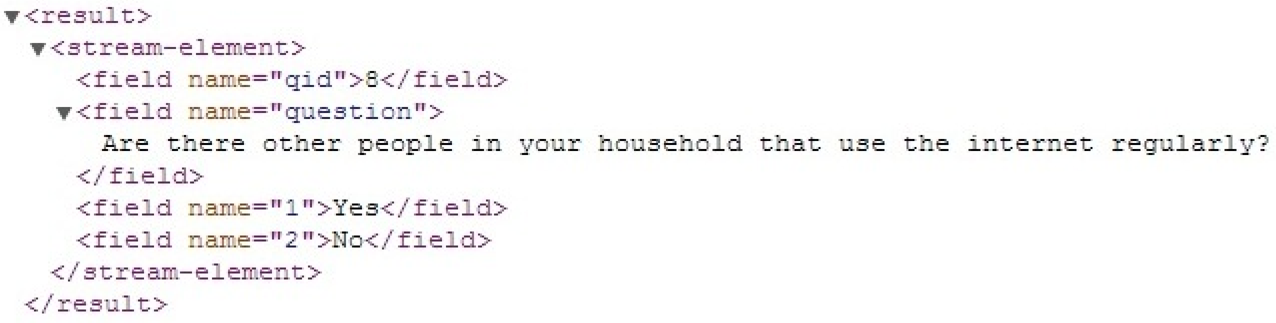
```
▼<result>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T20:00</field>
     <field name="VALUE">0.6145</field>
   </stream-element>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T20:30</field>
     <field name="VALUE">0.40650000000000003</field>
   </stream-element>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T21:00</field>
     <field name="VALUE">0.47950000000000004</field>
   </stream-element>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T21:30</field>
     <field name="VALUE">0.3935</field>
   </stream-element>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T22:00</field>
     <field name="VALUE">0.5505</field>
   </stream-element>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T22:30</field>
     <field name="VALUE">0.6035</field>
   </stream-element>
  ▼<stream-element>
     <field name="TIMESTAMP">2009-07-19T23:00</field>
     <field name="VALUE">0.34650000000000003</field>
   </stream-element>
 </result>
```

Figure 1: Example

Example:

http://localhost:22001/gsn?REQUEST=120&questions=1,1;60,2;5,1;3,2;34,1

Result: Figure 3.



```
▼<result>
  ▼<stream-element>
     <field name="qid">8</field>
    ▼<field name="question">
       Are there other people in your household that use the internet regularly?
     </field>
     <field name="1">Yes</field>
     <field name="2">No</field>
   </stream-element>
 </result>
```

Figure 2: Example

`REQUEST_SMARTD_QUESTION_SELECTION (http://localhost:22001/gsn?REQUEST=121)`

This request is handled by the class gsn.vsensor.http.SmartDQuestionSelection. This servlet executes a query over the GSN's internal tables and returns top 5 ranked question ids in the form of the stream elements. This request needs the following parameters:

- day : sunday,monday,tuesday,wednesday,thursday,friday,saturday,weekdays or weekend.
- season : winter,spring,summer,autumn.

Example:

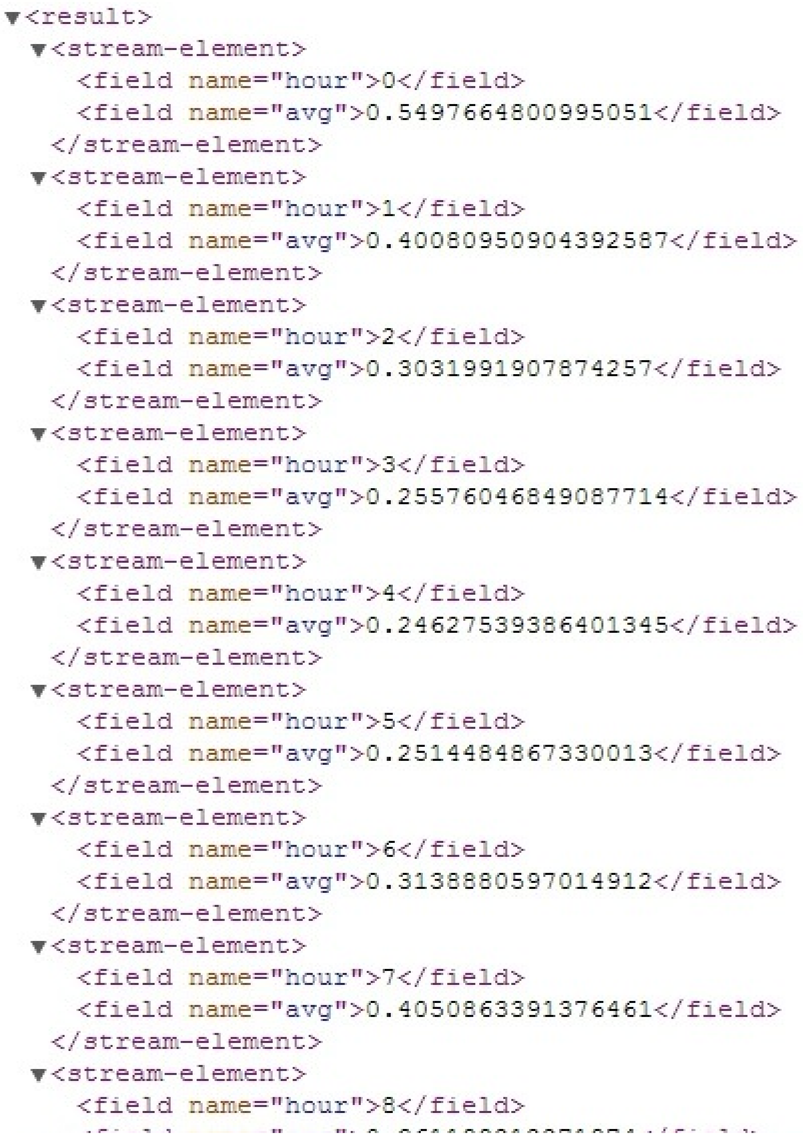
http://localhost:22001/gsn?REQUEST=121&day=weekdays&season=winte

Result: Figure 4.

`REQUEST_SMARTD_NEIGHBOR_NUM_EVALUATION (http://localhost:22001/gsn?REQUEST=122)`

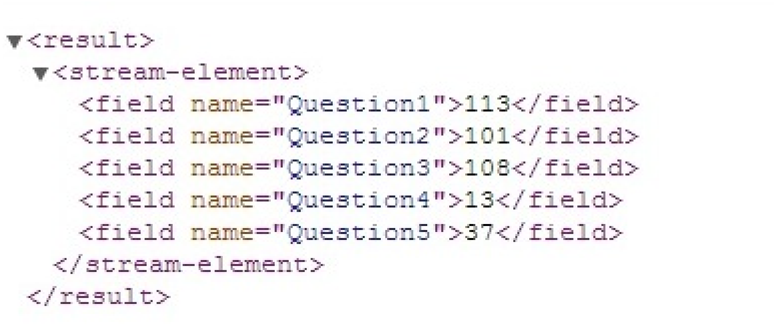
This request is handled by the class gsn.vsensor.http.SmartDNeighborNumEvaluation. This request needs the following parameter:

- season : all,winter,spring,summer,autumn.
- neighborNum



```
▼<result>
  ▼<stream-element>
     <field name="hour">0</field>
     <field name="avg">0.5497664800995051</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">1</field>
     <field name="avg">0.40080950904392587</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">2</field>
     <field name="avg">0.3031991907874257</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">3</field>
     <field name="avg">0.25576046849087714</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">4</field>
     <field name="avg">0.24627539386401345</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">5</field>
     <field name="avg">0.2514484867330013</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">6</field>
     <field name="avg">0.3138880597014912</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">7</field>
     <field name="avg">0.4050863391376461</field>
   </stream-element>
  ▼<stream-element>
     <field name="hour">8</field>
```

Figure 3: Example



```
▼<result>
  ▼<stream-element>
     <field name="Question1">113</field>
     <field name="Question2">101</field>
     <field name="Question3">108</field>
     <field name="Question4">13</field>
     <field name="Question5">37</field>
   </stream-element>
 </result>
```

Figure 4: Example

- questionNum

Example:

http://localhost:22001/gsn?REQUEST=122&neighborNum=20&questionNum=10

Result: Figure 5.

`REQUEST_SMARTD_QUESTION_NUM_EVALUATION (http://localhost:22001/gsn?REQUEST=123)`

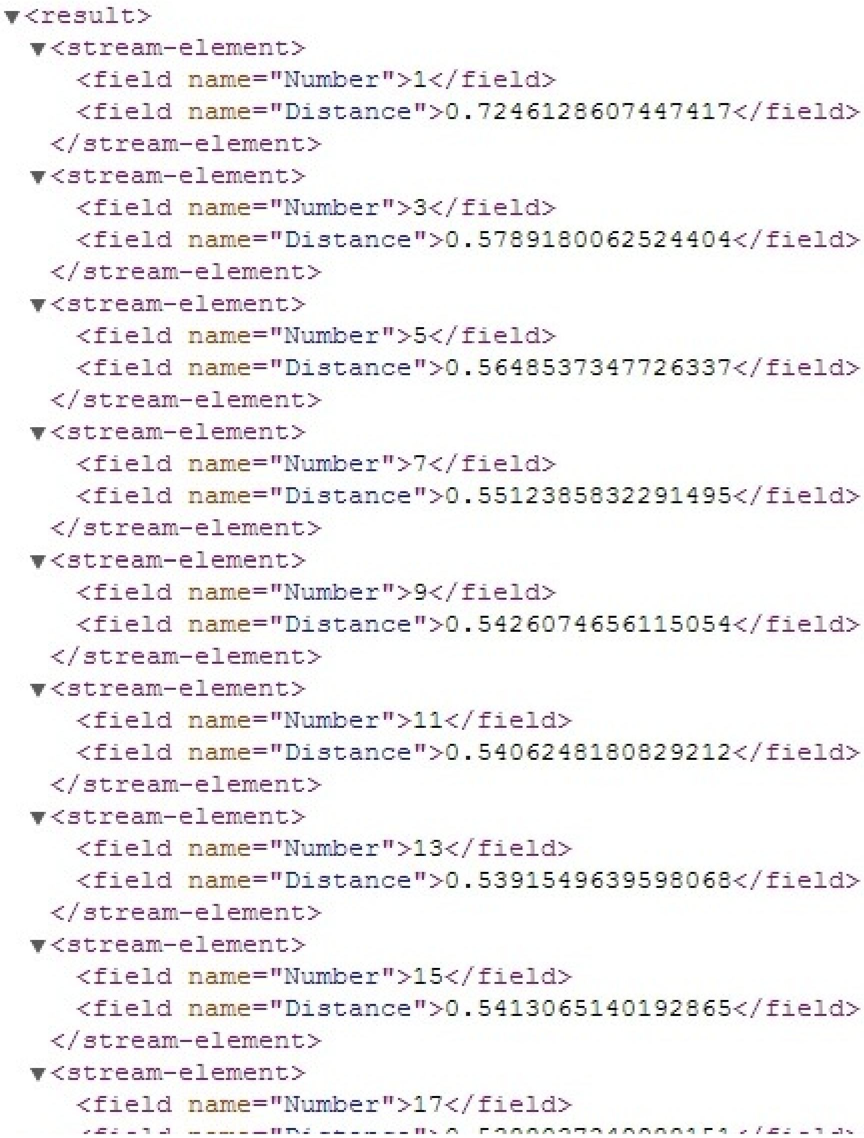
This request is handled by the class gsn.vsensor.http.SmartDQuestionNumEvaluaton. This request needs the following parameter:

- season : all,winter,spring,summer,autumn.
- neighborNum
- questionNum

Example:

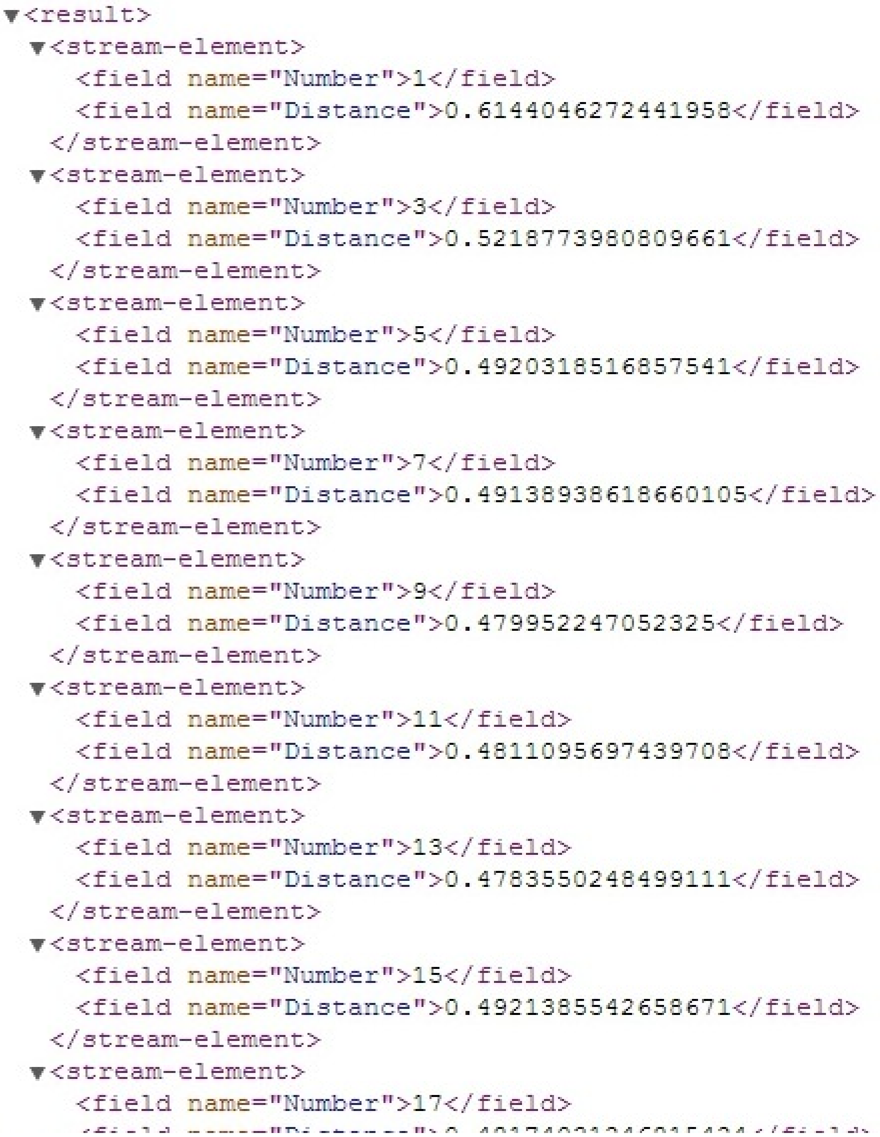
http://localhost:22001/gsn?REQUEST=123&neighborNum=20&questionNum=10

Result: Figure 6.



```
▼<result>
  ▼<stream-element>
     <field name="Number">1</field>
     <field name="Distance">0.7246128607447417</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">3</field>
     <field name="Distance">0.5789180062524404</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">5</field>
     <field name="Distance">0.5648537347726337</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">7</field>
     <field name="Distance">0.5512385832291495</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">9</field>
     <field name="Distance">0.5426074656115054</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">11</field>
     <field name="Distance">0.5406248180829212</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">13</field>
     <field name="Distance">0.5391549639598068</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">15</field>
     <field name="Distance">0.5413065140192865</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">17</field>
```

Figure 5: Example



```
▼<result>
  ▼<stream-element>
     <field name="Number">1</field>
     <field name="Distance">0.6144046272441958</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">3</field>
     <field name="Distance">0.5218773980809661</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">5</field>
     <field name="Distance">0.4920318516857541</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">7</field>
     <field name="Distance">0.49138938618660105</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">9</field>
     <field name="Distance">0.479952247052325</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">11</field>
     <field name="Distance">0.4811095697439708</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">13</field>
     <field name="Distance">0.4783550248499111</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">15</field>
     <field name="Distance">0.4921385542658671</field>
   </stream-element>
  ▼<stream-element>
     <field name="Number">17</field>
```

Figure 6: Example

`REQUEST_SMARTD_VALUE_DISTRIBUTION (http://localhost:22001/gsn?REQUEST=124)`

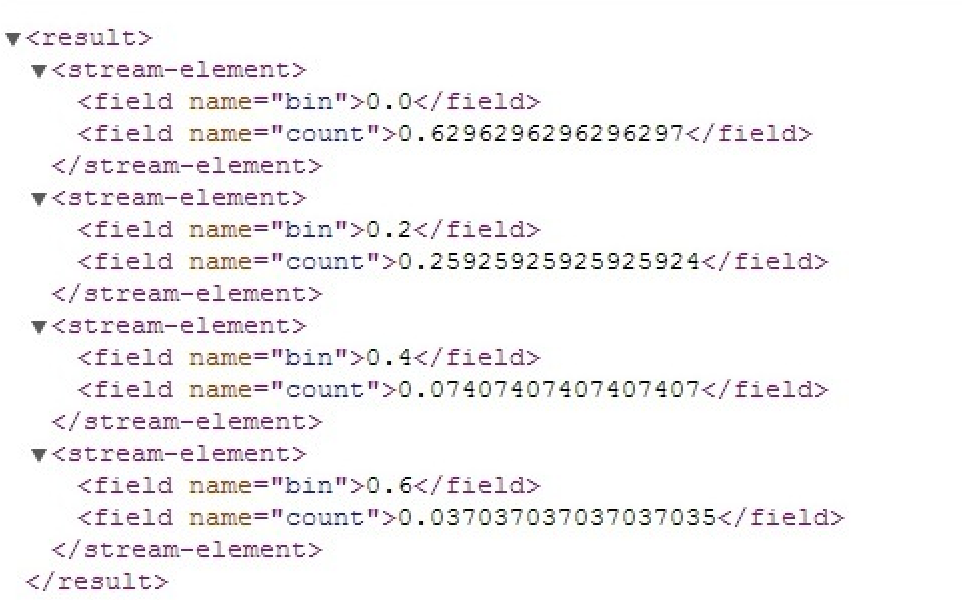
This request is handled by the class gsn.vsensor.http.SmartDValueDistribution. This request needs the following parameter:

- startTime
- endTime
- bin : binSize,binNum
- binInput
- id

Example:

http://localhost:22001/gsn?REQUEST=124&startTime=2009-07-19T01:00&endTime=2009-07-19T23:00&bin=binSize&binInput=0.2&id=1002

Result: Figure 7.



```
▼<result>
  ▼<stream-element>
     <field name="bin">0.0</field>
     <field name="count">0.6296296296296297</field>
   </stream-element>
  ▼<stream-element>
     <field name="bin">0.2</field>
     <field name="count">0.25925925925925924</field>
   </stream-element>
  ▼<stream-element>
     <field name="bin">0.4</field>
     <field name="count">0.07407407407407407</field>
   </stream-element>
  ▼<stream-element>
     <field name="bin">0.6</field>
     <field name="count">0.037037037037037035</field>
   </stream-element>
 </result>
```

Figure 7: Example