CineCubes: Cubes as Movie Stars with Little Effort

Dimitrios Gkesoulis*
UTC Creative Lab
Ioannina, Hellas

Panos Vassiliadis
Dept. of Computer Science & Engineering,
Univ. Ioannina, Hellas

*work conducted while in the Univ. Ioannina

Univ. of Ioannina
Can we answer user queries with more than just a set of records?

*We should and can produce query results that are*

- properly visualized,
- enriched with textual comments,
- vocally enriched,

*... but then, you have a movie*
Example

• Find the average work hours per week //measure

  – For persons with //selection conditions
    • work_class.level2=’With-Pay’ , and
    • education.level3= ‘Post-Sec’

  – Grouped per //groupers
    • work_class.level1
    • education.level3
Example

**Workclass.*

- With pay
  - Private
    - not-inc
  - Self-emp
  - Gov
  - Federal
  - Local
  - State
  - W/O pay

**Education.*

- W/O post secondary
  - Preschool
  - Elementary
    - 1st - 4th
    - 5th - 6th
    - 7th - 8th
  - Secondary
    - Junior secondary
    - Senior secondary
    - 9th
    - 10th
    - 11th
    - 12th
    - HS-grad
  - Some college
  - Assoc
  - Univ
  - Post grad
  - Post - secondary
  - adm
  - voc
  - BSc
  - Prof
  - MSc
  - PhD

L3
L2
L1
L0
L4
L3
L1
L0
Answer to the original question

<table>
<thead>
<tr>
<th></th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td>40.73</td>
<td>43.58</td>
<td>38.38</td>
<td>42.14</td>
</tr>
<tr>
<td>Private</td>
<td>41.06</td>
<td>45.19</td>
<td>38.73</td>
<td>43.06</td>
</tr>
<tr>
<td>Self-emp</td>
<td>46.68</td>
<td>47.24</td>
<td>45.70</td>
<td>46.61</td>
</tr>
</tbody>
</table>

Here, you can see the answer of the original query. You have specified education to be equal to 'Post-Secondary', and work to be equal to 'With-Pay'. We report on Avg of work hours per week grouped by education at level 2, and work at level 1. You can observe the results in this table. We highlight the largest values with red and the lowest values with blue color. Column Some-college has 2 of the 3 lowest values. Row Self-emp has 3 of the 3 highest values. Row Gov has 2 of the 3 lowest values.
## Drilling down education

<table>
<thead>
<tr>
<th>Assoc</th>
<th>Gov</th>
<th>Private</th>
<th>Self-emp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assoc-acm</td>
<td>39.91(182)</td>
<td>40.87(720)</td>
<td>45.49(105)</td>
</tr>
<tr>
<td>Assoc-voc</td>
<td>41.61(169)</td>
<td>41.20(993)</td>
<td>47.55(145)</td>
</tr>
<tr>
<td>Post-grad</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Doctorate</td>
<td>46.53(124)</td>
<td>49.05(172)</td>
<td>47.22(79)</td>
</tr>
<tr>
<td>Masters</td>
<td>42.93(567)</td>
<td>44.42(863)</td>
<td>47.25(197)</td>
</tr>
<tr>
<td>Some-college</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Some-college</td>
<td>38.38(955)</td>
<td>38.73(5016)</td>
<td>45.70(704)</td>
</tr>
<tr>
<td>University</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bachelors</td>
<td>41.56(943)</td>
<td>42.71(3455)</td>
<td>46.23(646)</td>
</tr>
<tr>
<td>Prof-school</td>
<td>48.40(86)</td>
<td>47.96(247)</td>
<td>47.78(209)</td>
</tr>
</tbody>
</table>

## Summary for education

### Act I (sl. 2,3)

In this slide, we drill-down one level for all values of dimension work at level 0. For each cell we show both the Avg of Hrs and the number of tuples that correspond to it in parentheses. 
- Column Post-grad has 4 of the 6 highest values.
- Column Some-college has 4 of the 6 lowest values.

### Drilling down work

<table>
<thead>
<tr>
<th>Gov</th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Federal-gov</td>
<td>41.15(93)</td>
<td>43.86(80)</td>
<td>40.31(251)</td>
<td>43.38(233)</td>
</tr>
<tr>
<td>Local-gov</td>
<td>41.33(171)</td>
<td>43.96(362)</td>
<td>40.14(385)</td>
<td>42.34(499)</td>
</tr>
<tr>
<td>State-gov</td>
<td>39.09(87)</td>
<td>42.93(249)</td>
<td>34.73(319)</td>
<td>40.82(297)</td>
</tr>
<tr>
<td>Private</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Private</td>
<td>41.06(1713)</td>
<td>45.19(1035)</td>
<td>38.73(5016)</td>
<td>43.06(3702)</td>
</tr>
<tr>
<td>Self-emp</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Self-emp-inc</td>
<td>48.68(72)</td>
<td>53.05(110)</td>
<td>49.31(223)</td>
<td>49.91(338)</td>
</tr>
<tr>
<td>Self-emp-not-inc</td>
<td>45.88(178)</td>
<td>43.39(166)</td>
<td>44.03(481)</td>
<td>44.44(517)</td>
</tr>
</tbody>
</table>

## Summary for work

In this graphic, we put the original request in context by comparing the value 'Post-Secondary' for education at level 3 with its sibling values. We calculate the Avg of Hrs while fixing education at level 4 to be equal to "ALL", and work at level 2 to be equal to "With-Pay". We highlight the reference cells with bold, the highest value with red and the lowest value with blue.

Compared to its sibling, we observe that in 3 out of 3 cases Post-Secondary has higher value than Without-Post-Secondary.
Contributions

• We create a small “movie” that answers an OLAP query
• We complement each query with auxiliary queries organized in thematically related acts that allow us to assess and explain the results of the original query
• We implemented an extensible palette of highlight extraction methods to find interesting patterns in the result of each query
• We describe each highlight with text
• We use TTS technology to convert text to audio
Contributions

• Equally importantly:
  – An extensible software where algorithms for query generation and highlight extraction can be plagued in
  – The demonstration of low technical barrier to produce CineCube reports
Method Overview
Software Issues
Discussion

Method Overview
Our Approach

• A first assessment of the current state of affairs
  – Practically, this requirement refers to the execution of the original query.
• Put the state in Context
• Analysis of why things are this way.
Structure of the CineCube Movie

• A typical movie story is structured in acts.
  – Each Act is composed of sequences of scenes

• That’s why we organize the CineCube Movie in five Acts:
  – Intro Act
  – Original Act
  – Act I
  – Act II
  – Summary Act
The movie’s parts

- Much like movies, we organize our stories in acts
- Each act including several episodes all serving the same purpose
CineCube Movie – Intro Act

• Intro Act has an episode that introduce the story to user
CineCube Movie – Original Act

- Original Act has an episode which is the answer of query that submitted by user.
CineCube Movie – Act I

• In this Act we try to answer the following question:
  – How good is the original query compared to its siblings?

• We compare the marginal aggregate results of the original query to the results of “sibling” queries that use “similar” values in their selection conditions
Act I – Example

Result of Original Query

<table>
<thead>
<tr>
<th></th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td>40.73</td>
<td>43.58</td>
<td>38.38</td>
<td>42.14</td>
</tr>
<tr>
<td>Private</td>
<td>41.06</td>
<td>45.19</td>
<td>38.73</td>
<td>43.06</td>
</tr>
<tr>
<td>Self-emp</td>
<td>46.68</td>
<td>47.24</td>
<td>45.70</td>
<td>46.61</td>
</tr>
</tbody>
</table>

\[ q = (DS^0, W.L_2 = 'With-Pay' \land E.L_3 = 'Post-Sec', [W.L_1, E.L_2], \text{avg}(\text{Hrs})) \]

Assessing the behavior of education

<table>
<thead>
<tr>
<th>Summary for education</th>
<th>Post-Secondary</th>
<th>Without-Post-Secondary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td>41.12</td>
<td>38.97</td>
</tr>
<tr>
<td>Private</td>
<td>41.06</td>
<td>39.40</td>
</tr>
<tr>
<td>Self-emp</td>
<td>46.39</td>
<td>44.84</td>
</tr>
</tbody>
</table>

\[ q = (DS^0, W.L_2 = 'With-Pay' \land E.L_4 = 'All', [W.L_1, E.L_3], \text{avg}(\text{Hrs})) \]
### Act I – Example

#### Result of Original Query

<table>
<thead>
<tr>
<th></th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td>40.73</td>
<td>43.58</td>
<td><strong>38.38</strong></td>
<td>42.14</td>
</tr>
<tr>
<td>Private</td>
<td>41.06</td>
<td>45.19</td>
<td><strong>38.73</strong></td>
<td>43.06</td>
</tr>
<tr>
<td>Self-emp</td>
<td><strong>46.68</strong></td>
<td><strong>47.24</strong></td>
<td><strong>45.70</strong></td>
<td><strong>46.61</strong></td>
</tr>
</tbody>
</table>

$q=(DS^0, W.I_2 = 'With-Pay' \land E.I_3 = 'Post-Sec', [W.L_1, E.L_2], \text{avg}(Hrs))$

### Assessing the behavior of work

#### Summary for work

<table>
<thead>
<tr>
<th></th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>With-Pay</td>
<td>41.62</td>
<td>44.91</td>
<td><strong>39.41</strong></td>
<td><strong>43.44</strong></td>
</tr>
<tr>
<td>Without-pay</td>
<td>50.00</td>
<td>-</td>
<td><strong>35.33</strong></td>
<td>-</td>
</tr>
</tbody>
</table>

$q=(DS^0, W.I_3 = 'All' \land E.I_3 = 'Post-Sec', [W.L_2, E.L_2], \text{avg}(Hrs))$

---

![Tree Diagram]
CineCube Movie – Act II

• In this Act we try to explaining to user **why** the result of original query is what it is.
  – “Drilling into the breakdown of the original result”

• We drill in the details of the cells of the original result in order to inspect the internals of the aggregated measures of the original query.
## Act II – Example

### Result of Original Query

<table>
<thead>
<tr>
<th></th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td>40.73</td>
<td>43.58</td>
<td><strong>38.38</strong></td>
<td>42.14</td>
</tr>
<tr>
<td>Private</td>
<td>41.06</td>
<td>45.19</td>
<td><strong>38.73</strong></td>
<td>43.06</td>
</tr>
<tr>
<td>Self-emp</td>
<td><strong>46.68</strong></td>
<td><strong>47.24</strong></td>
<td><strong>45.70</strong></td>
<td><strong>46.61</strong></td>
</tr>
</tbody>
</table>

\[
q = (DS^0, \quad W.I_{2} = 'With-Pay' \land E.I_{3} = 'Post-Sec', \quad [W.I_1, E.I_2], \quad \text{avg(Hrs)})
\]

### Drilling down the Rows of the Original Result

<table>
<thead>
<tr>
<th></th>
<th>Assoc</th>
<th>Post-grad</th>
<th>Some-college</th>
<th>University</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Federal-gov</td>
<td>41.15 (93)</td>
<td>43.86 (80)</td>
<td>40.31 (251)</td>
<td>43.38 (233)</td>
</tr>
<tr>
<td>Local-gov</td>
<td>41.33 (171)</td>
<td>43.96 (362)</td>
<td>40.14 (385)</td>
<td>42.34 (499)</td>
</tr>
<tr>
<td>State-gov</td>
<td><strong>39.09 (87)</strong></td>
<td>42.93 (249)</td>
<td><strong>34.73 (319)</strong></td>
<td>40.82 (297)</td>
</tr>
<tr>
<td>Private</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Private</td>
<td>41.06 (1713)</td>
<td>45.19 (1035)</td>
<td>38.73 (5016)</td>
<td>43.06 (3702)</td>
</tr>
<tr>
<td>Self-emp</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Self-emp-inc</td>
<td>48.68 (72)</td>
<td>53.05 (110)</td>
<td>49.31 (223)</td>
<td>49.91 (338)</td>
</tr>
<tr>
<td>Self-emp-not-inc</td>
<td>45.88 (178)</td>
<td>43.39 (166)</td>
<td>44.03 (481)</td>
<td>44.44 (517)</td>
</tr>
</tbody>
</table>
### Act II – Example

#### Result of Original Query

| Assoc            | Post-grad | Some-college | University | q = \( DS^0, \
W.I_2 = 'With-Pay' \land E.I_3 = 'Post-Sec', [W.L_1, E.L_2], \\text{avg(Hrs)} \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Gov</td>
<td>40.73</td>
<td>43.58</td>
<td>42.14</td>
<td></td>
</tr>
<tr>
<td>Private</td>
<td>41.06</td>
<td>45.19</td>
<td>43.06</td>
<td></td>
</tr>
<tr>
<td>Self-emp</td>
<td>46.68</td>
<td>47.24</td>
<td>46.61</td>
<td></td>
</tr>
</tbody>
</table>

#### Drilling down the Columns of the Original Result

<table>
<thead>
<tr>
<th>Assoc</th>
<th>Gov</th>
<th>Private</th>
<th>Self-emp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assoc-acdm</td>
<td>39.91 (182)</td>
<td>40.87 (720)</td>
<td>45.49 (105)</td>
</tr>
<tr>
<td>Assoc-voc</td>
<td>41.61 (169)</td>
<td>41.20 (993)</td>
<td>47.55 (145)</td>
</tr>
<tr>
<td>Post-grad</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Doctorate</td>
<td>46.53 (124)</td>
<td>49.05 (172)</td>
<td>47.22 (79)</td>
</tr>
<tr>
<td>Masters</td>
<td>42.93 (567)</td>
<td>44.42 (863)</td>
<td>47.25 (197)</td>
</tr>
<tr>
<td>Some-college</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Some-college</td>
<td>38.38 (955)</td>
<td>38.73 (5016)</td>
<td>45.70 (704)</td>
</tr>
<tr>
<td>University</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bachelors</td>
<td>41.56 (943)</td>
<td>42.71 (3455)</td>
<td>46.23 (646)</td>
</tr>
<tr>
<td>Prof-school</td>
<td>48.40 (86)</td>
<td>47.96 (247)</td>
<td>47.78 (209)</td>
</tr>
</tbody>
</table>
CineCube Movie – Summary Act

• Summary Act represented from one episode.
• This episode has all the highlights of our story.
Highlight Extraction

• We utilize a palette of highlight extraction methods that take a 2D matrix as input and produce important findings as output.

• Such as:
  – The top and bottom quartile of values in a matrix
  – The absence of values from a row or column
  – The domination of a quartile by a row or a column
  – The identification of min and max values
Text Extraction

• Text is constructed by a Text Manager that customizes the text per Act

• Example:

   *In this slide, we drill-down one level for all values of dimension* <dim> *at level* <l>. *For each cell we show both the* <agg> *of* <measure> *and the number of tuples that correspond to it.*
Experimental Results

Time breakdown (msec, log scale) for the method’s parts

- Result Generation
- Highlight Extraction & Visualization
- Text Creation
- Audio Creation
- Put in PPTX
Software Issues
Low technical barrier

• Our tool is extensible
  – We can add new tasks to generate complementary queries easily
  – We can add new highlight algorithms to produce highlights easily

• Supportive technologies are surprisingly easier to use
  – Apache POI for pptx generation
  – TTS for text to speech conversion
Apache POI for pptx

- A Java API that provides several libraries for Microsoft Word, PowerPoint and Excel (since 2001).
- XSLF is the Java implementation of the PowerPoint 2007 OOXXML (.pptx) file format.

```java
XMLSlideShow ss = new XMLSlideShow();
XSLFFSlideMaster sm = ss.getSlideMasters()[0];

XSLFFSlide sl = ss.createSlide(sm.getLayout(SlideLayout.TITLE_AND_CONTENT));

XSLFTable t = sl.createTable();
t.addRow().addCell().setText("added a cell");
```
MaryTTS for Text-to-Speech Synthesis

MaryInterface m = new LocalMaryInterface();
m.setVoice("cmu-slt-hsmm");

AudioInputStream audio = m.generateAudio("Hello");

AudioSystem.write(audio, audioFileFormat.Type.WAVE, new File("myWav.wav"));
Method Overview
Software Issues
Discussion

Discussion
Open issues

- Multi-query
- Speed-up voice gen.
- Cloud/parallel
- Back stage
- Show text
- More than 2D arrays
- Visualize
- Look like a movie
- CC now
- Info content
- Structure more like a movie
- More highlights
- Crowd wisdom
- More acts (more queries)
- Chase after interestingness

- Interacti on
- How to allow interaction with the user?
- Personalization

- Assumptions
- 2D results (2 groupers)
- Star schema
- Equality selections
- Single measure

Be compendious; if not, at least be concise!
Thank you!

Any questions?

More information


Demo


Code

- [https://github.com/DAINTINESS-Group/CinecubesPublic.git](https://github.com/DAINTINESS-Group/CinecubesPublic.git)
AUXILIARY SLIDES
Related Work
Related Work

• Query Recommendations
• Database-related efforts
• OLAP-related methods
• Advanced OLAP operators
• Text synthesis from query results
Query Recommendations


Database-related efforts


OLAP-related methods


Advanced OLAP operators


• S. Sarawagi, 1999. Explaining Differences in Multidimensional Aggregates. VLDB (Edinburgh, Scotland, 1999), pp. 42-53

• G. Sathe, S. Sarawagi, 2001. Intelligent Rollups in Multidimensional OLAP Data. VLDB (Roma, Italy 2001), pp.531-540
Text synthesis from query results

Formalities
OLAP Model

- We base our approach on an OLAP model that involves
  - Dimensions, defined as lattices of dimension levels
  - Ancestor functions, (in the form of $\text{anc}_{L_1}^{L_2}$) mapping values between related levels of a dimension
  - Detailed data sets, practically modeling fact tables at the lowest granule of information
  - Cubes, defined as aggregations over detailed data sets
What is Cube?

- A primary Cube $C$ is described as
  \[ C = (DS^0, \Phi, [L_1, \ldots, L_n, M_1, \ldots, M_m], [agg_1(M_1^0), \ldots, agg_1(M_m^0)]) \]

- $DS^0$ is a detailed dataset over the schema
- $\Phi$ is a detailed selection condition
  - $\Phi$ analyzed as $\varphi_1 \land \cdots \land \varphi_k$
  - $\varphi_i$ is $D_i.L_j = value_i$
- $L_1, \ldots, L_n$ are levels such that $L_i < L_{i+1}, 1 \leq i \leq n.$
- $M_1, \ldots, M_m$ are measures
- $agg_i \in \{max, min, sum, count, average\}, 1 \leq i \leq m$
Cube Query

- A cube query \( Q \) can be considered as \( Q = (\mathcal{DS}^0, \Sigma, \Gamma, \gamma(M)) \)

- where:
  - \( \Sigma \) is a conjunction of dimensional restrictions of the form
  - \( \Gamma \) is a set of grouper dimensional level
  - \( \gamma(M) \) is an aggregate function applied to the measure of the cube
Cube Query

- In our approach we assume that the user submit cube queries which denote as:
  - $q=(D^0, \varphi_1 \land \cdots \land \varphi_k, [L_\alpha, L_\beta], \text{agg}(M))$

- Example:

  $q=(A, W.L_2 = 'With-Pay' \land E.L_3 = 'Post-Sec', [W.L_1, E.L_2], \text{avg}(Hrs))$
In general case:

\[
\text{SELECT } L_1, \ldots, L_n, \text{agg}_1(M_1^0), \ldots, \text{agg}_1(M_1^0)]] \\
\text{FROM } DS^0 \text{ INNER JOIN } D_1, \ldots \text{INNER JOIN } D_n \\
\text{WHERE } \phi \\
\text{GROUP BY } L_1, \ldots, L_n
\]

Example for our case:

\[
\text{SELECT } W.L_1, E.L_2, \text{AVG}(Hrs) \\
\text{FROM } A \\
\text{INNER JOIN } W \text{ ON } A.W = W.L_0 \\
\text{INNER JOIN } E \text{ ON } A.E = E.L_0 \\
\text{WHERE } W.L_2 = 'With-Pay' \text{ AND } E.L_3 = 'Post-Sec' \\
\text{GROUP BY } W.L_1, E.L_2
\]
Method Internals
Act I – Our Definition

- We introduce **two marginal sibling queries**, one for each aggregator.
- Formally, given an original query:
  \[ q = (D S^0, \varphi_1 \wedge \ldots \wedge \varphi_k, [L_\alpha, L_\beta], \text{agg}(M)) \]
- Its two marginal sibling queries are:
  1. \[ q^s = (D S^0, \varphi_1 \wedge \ldots \varphi_\chi^* \wedge \ldots \wedge \varphi_k, [L_\alpha, L_\chi], \text{agg}(M)) \]
  2. \[ q^s = (D S^0, \varphi_1 \wedge \ldots \varphi_\chi^* \wedge \ldots \wedge \varphi_k, [L_\chi, L_\beta], \text{agg}(M)) \]
    - \( \varphi_\chi^*: L_{x+1} = \text{anc}_{L_{x+1}}(v) \)
Act I – Query Example

- **Original Query**
  \[ q = (DS^0, W.L_2 = 'With-Pay' \land E.L_3 = 'Post-Sec', [W.L_1, E.L_2], \text{avg(Hrs)}) \]

- **Sibling Queries:**
  1. \[ q = (DS^0, W.L_2 = 'With-Pay' \land E.L_4 = 'All', [W.L_1, E.L_3], \text{avg(Hrs)}) \]
  2. \[ q = (DS^0, W.L_3 = 'All' \land E.L_3 = 'Post-Sec', [W.L_2, E.L_2], \text{avg(Hrs)}) \]
Act I – How produce it?

- We define a sibling query as a query with a single difference to the original:
  - Instead of an atomic selection formula $L_i = v_i$, the sibling query contains a formula of the form $L_i \in \text{children}(\text{parent}(v_i))$.

- Formally, given an original query

\[
q = (DS^0, \varphi_1 \land \cdots \land \varphi_k, [L_\alpha, L_\beta], \text{agg}(M))
\]

- A new query $q^s$ is a sibling query if is of the form

\[
q^s = (DS^0, \varphi_1 \land \cdots \varphi_\chi^* \land \cdots \land \varphi_k, [L_\alpha, L_\beta], \text{agg}(M))
\]

  - $\varphi_\chi^*: L_{x+1} = \text{anc}_{L_x}^{L_{x+1}}(v)$
Act II – Query Example

- **Original Query**
  - \( q = (DS^0, W.L_2 = 'With-Pay' \land E.L_3 = 'Post-Sec', [W.L_1, E.L_2], \text{avg}(Hrs)) \)

- **Drill in Queries for work dimension:**
  1. \( q = (DS^0, W.L_1 = 'Gov' \land E.L_3 = 'Post-Sec', [W.L_0, E.L_2], \text{avg}(Hrs)) \)
  2. \( q = (DS^0, W.L_1 = 'Private' \land E.L_3 = 'Post-Sec', [W.L_0, E.L_2], \text{avg}(Hrs)) \)
  3. \( q = (DS^0, W.L_1 = 'Self-emp' \land E.L_3 = 'Post-Sec', [W.L_0, E.L_2], \text{avg}(Hrs)) \)

For Education dimension: similarly
Act II- How produce it?

- Assume a cube query and its result, visualized as a 2D matrix.
- For each cell $c$ of this result is characterized by the following cube query:
  - $q^c = (DS^0, \phi_1 \land \ldots \land \phi_k \land \phi_c, [L_\alpha, L_\beta], \text{agg}(M))$
  - $\varphi_c : L_\alpha = v^c_a \land L_\beta = v^c_\beta$
Act II- How produce it?

- For each of the aggregator dimensions, we can generate a set of explanatory drill in queries, one per value in the original result:

1. \( q^s_a = (DS^0, \phi_1 \land \ldots \land \phi_k \land \phi, [L_{\alpha-1}, L_{\beta}], \text{agg}(M)) \),

2. \( q^s_\beta = (DS^0, \varphi_1 \land \ldots \land \varphi_k \land \varphi_c, [L_\alpha, L_{\beta-1}], \text{agg}(M)) \)

\[ \cdot \varphi_c : L_\alpha = \nu^c_\alpha \land L_\beta = \nu^c_\beta \]
Our Algorithm

Algorithm Construct Operational Act

Input: the original query over the appropriate database

Output: a set of an act’s episodes fully computed

1. Create the necessary objects (act, episodes, tasks, subtasks) appropriately linked to each other
2. Construct the necessary queries for all the subtasks of the Act, execute them, and organize the result as a set of aggregated cells (each including its coordinates, its measure and the number of its generating detailed tuples)
3. For each episode
   - Calculate the visual presentation of cells
   - Calculate the cells’ highlights
   - Produce the text based on the highlights
   - Produce the audio based on the text
Experiments and Results
Experimental setup

• Adult dataset referring to data from 1994 USA census
  – Has 7 dimension Age, Native Country, Education, Occupation, Marital status, Work class, and Race.
  – One Measure : work hours per week

• Machine Setup :
  – Running Windows 7
  – Intel Core Duo CPU at 2.50GHz,
  – 3GB main memory.
Experiments

Time breakdown (msec) per Act

- Intro Act
- Original Act
- Act I
- Act II
- Summary Act