Askia Data structure

As you know the current Askia Data structure creates a lot of tables and does not deal well with multiple questions with a lot of responses. Each grid creates at least one table… a multiple question with 160 responses would need 2 tables.

One of the other problems is the fact that any change to the questionnaire usually means a verification of the data structure… blocking any access to the database and a very fragile process because of its complexity.

Most databases deal well with queries over few tables so we are proposing the following data structure

# New structure

I will describe the proposal and detail below the possible variations… as usual the devil is in the detail.

We have 2 tables – one table for logging all the peri-data or interview data… that’s the start and end of interview, IP address, geographical location, seed etc… and one table for the case data that is the responses to the questions (and time spent doing so).

#### Interview table – usually called AskiaNNNInterview where NNN is the id of the survey in askiaField.

Some fields have kept the same names as before ( like CodeEnqueteur) which I do not believe is a good idea.

|  |  |  |
| --- | --- | --- |
| **InterviewNN** | **Type** | **Comment** |
| InterviewID | AutoNumber |  |
| AgentID | Identifier (Number) |  |
| StartInterview | Date |  |
| EndInterview | Date |  |
| Seed | Number |  |
| CallId | Identifier (Number) |  |
| IPAddress | String |  |
| Completed | Boolean |  |
| LanguageId | Identifier (Number) |  |
| Quotas | String | Need re-engineering |
| QuotasToDo | String | Need re-engineering |
| KeyStrokeCount | Number | Probably obsolete |
| ErrorsOnVerifyCount | Number | Probably obsolete |
| Guid | String |  |
| LastResult | Number |  |
| LastSubResult | Number |  |
| CallCount | Number |  |
| Revision | Number |  |
| … more to come |  |  |

#### Data table – usually called AskiaNNNData where NNN is the id of the survey in askiaField.

|  |  |  |
| --- | --- | --- |
| DataNN | **Type** | **Comment** |
| InterviewID | Identifier (Number) |  |
| LoopID1 | Identifier (Number) |  |
| LoopID2 | Identifier (Number) |  |
| LoopID3 | Identifier (Number) |  |
| … |  |  |
| QuestionID | Identifier (Number) |  |
| DataOrder | Number | Use for multiple questions |
| Duration | Number |  |
| ClosedData | Identifier (Number) |  |
| OpenData | String |  |
| NumericData | Double |  |

Depending on the question type, only one of the ClosedData or OpenData or NumericData field would be filled (or none for a chapter).

The LoopIDNN fields indicate in which loop a question is… the number of these fields should be the maximum depth of imbricated loops but we could leave it at 2 by default to avoid 95% of problems and only increase it when we need it.

Let’s give an example:

Q1 = Gender (male = 1, female = 2)

Q2 = Name (open)

Q3 = Age (numeric)

Q4 = Newspapers (multiple) ( Guardian = 3,….,Times =7, … Daily bugle =9)

Loop5 = Sentences (question table, 1st sentence = 20, 2nd = 21, 3rd = 22)

-> Q6 = Agree with sentences (Agrees a lot=10, Agrees a bit = 11, Disagrees=12, Disagrees a lot=22)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| InterviewID | LoopID1 | QuestionID | DataOrder | Duration | ClosedData | NumericData | OpenData | Comment |
| 1 |  | 1 | 1 | 3 | 2 |  |  | 2 is the ModalityID for female |
| 1 |  | 2 | 1 | 4 |  | 25 |  | Age is 25 |
| 1 |  | 3 | 1 | 4 |  |  | Mary | Name is Mary |
| 1 |  | 4 | 1 | 5 | 3 |  |  | First newspaper is Guardian |
| 1 |  | 4 | 2 |  | 7 |  |  | Second newspaper is Times (no duration on following multiple) |
| 1 |  | 4 | 3 |  | 9 |  |  | Third newspaper is Daily Bugle |
| 1 |  | 5 | 1 |  | 20 |  |  | Sentence 1 was shown first (Modality ID of Sentence1 is 20) |
| 1 |  | 5 | 2 |  | 21 |  |  | Sentence 2 was shown second |
| 1 |  | 5 | 3 |  | 22 |  |  | Sentence 3 was shown third |
| 1 | 20 | 6 | 1 | 2 | 10 |  |  | Agrees with 1st sentence |
| 1 | 21 | 6 | 1 | 3 | 11 |  |  | Agrees a bit 2nd sentence |
| 1 | 22 | 6 | 1 | 5 |  |  |  | Disagrees a lot with 3rd sentence |

#### Conclusion

Reading a whole interview will only take one SELECT… but writing an interview will probably need more INSERTs than before. But to make the CCA more responsive, it’s the reading we need to improve no the writing… so that’s not a bad thing.

There will be no more “verification of the data structure”… and none of the uncertainties associated with it… Multiple questions will be supported much more elegantly and efficiently.

The whole database will be marginally bigger… but much easier to maintain / archive or query.