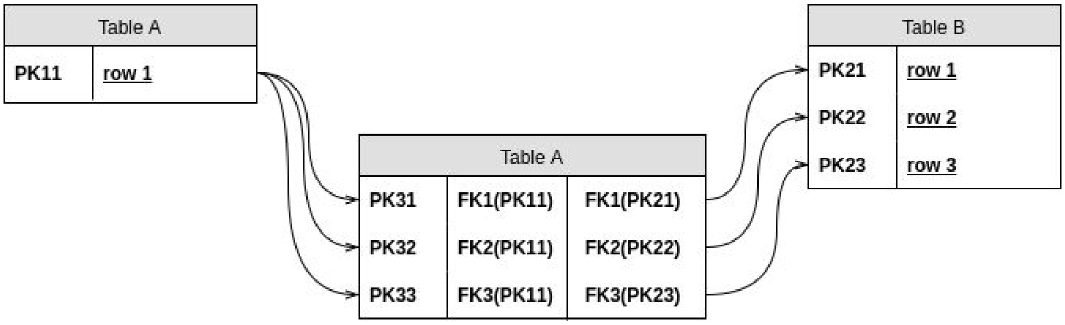
Foreign keys for Array elements

I am writing this proposal in reference to the third project idea in the wikia page.

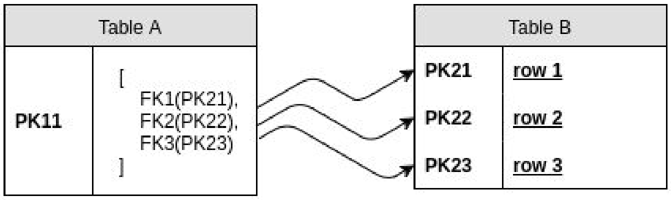
Benefits to the PostgreSQL Community

think that when this feature is completed it would be of great benefit to database designers in the open source community as no other DBMS offers it and there is a niche in the community since it is not part of the SQL standard.

For instance, if one wishes to create a many - many relationship between two tables he would have to create a third table to accommodate the references as the figure shows below.



However, a much simpler (and more intuitive approach) would be to include a foreign key array in Table A referencing the desired records found in Table B as shown below.



This by far closer to a programmer's mindset if he is still new to the database world.

This is only one example (and the first one that came to mind) where this new feature is useful, however, more benefits are bound to come out of this and I for one am excited how the open source community would take advantage of this feature. (I am sure I will be pleasantly surprised)

Deliverables

I understand that there were **three** previous patches targeting this specific feature. As it may seem simple in theory, but I am sure it will prove troublesome code-wise since many problems (performance issues & loopholes) will arise as I progress.

Skimming through the previous three patches (v1 Marco Nenciarini , v2 Marco Nenciarini, v3 Tom Lane) I have gathered that I should be concerned mainly with 21 files (excluding the 4 test files) for the development stage

Issues mentioned by Tom Lane:

1. The use of `count(distinct y)` in the SQL statements if the referencing column is an array. Since its equality operator is different from the PK unique index equality operator this leads to a broken statement[1]
2. Performance issues arise when performing an UPDATE or DELETE statements on the PK table, as it is unavoidable not to perform a full-table sequential scan on the FK table.2]

I am sure there are other issues, however, I think the second issue (beyond a shadow of a doubt) is the one that renders this feature unusable in the previous updates. Since it constraints the foreign key array feature to only work with relatively small tables.

I am aware that this project will involve more research than actual code writing.

Ultimately I plan to deliver a patch (v4 if I may) to resolve the second issue by the end of the summer

Project Schedule

Focusing on the full-table sequential scan issue.

The project is exactly twelve weeks long (or less) and this is how I plan to manage my time:

I would divide my time into three phases (Research, Development, Testing)

Research (two weeks):

1. My first order of business would be to get familiar with the PostgreSQL architecture. This should start to take place before the official start of the project (30th of May)
2. By **30th of May** I hope I have sufficient knowledge of the architecture to know my way around. I will then read the previous patches more extensively to know exactly where those who came before me have reached and learn from their efforts. I would give myself **two** weeks of research before commencing to code. I prefer this than a head on approach as it saves time later on.
3. Making the full-table sequential scan **GIN-indexable** instead seems very reasonable since GIN is primarily used to search for element values (PK values) that appear within composite items (FK array).
    a. Statistics have shown[7] that GIN indexing an array shows an increase in performance by ~2256% !
    b. Thus the first step (as proposed) would be to prove that “<@(is contained by)” can be used in this scope.

Development (seven weeks):

The development phase would become clearer once the research phase is done. I would like to postpone any development plans till then to make an informed decision.

The first phase ends on June 30th, the research phase would have presumably been over, and progress would be achieved. I will not come out with my hands empty by the end of the first phase.

Testing (three weeks):

Throughout development, I incrementally test my progress to ensure I on the right track. This is separate from the testing phase, where I run the previously written tests in the old patches and add some of my own to ensure I have successfully overcome the targeted performance issue.

Bio

My name is Mark Rofail, from Alexandria, Egypt. I am a third-year Computer Science and Engineering undergrad at the German University in Cairo, Egypt.

I would be delighted if you would check my CV[6].

I have been programming for 3 years (still considered a novice compared to the elites in the open source community). My main programming language is Java, but I have used several others in my projects such as C, Javascript, PHP, Haskell, Prolog. Through these 3, years I have worked on 30+ projects (academic and side projects).

My experience with Database Systems:

- Data Bases I (CSEN 501)[3]

I have been programming for 3 years (still considered a novice compared to the elites in the open source community). My main programming language is Java, but I have used several others in my projects such as C, Javascript, PHP, Haskell, Prolog. Through these 3, years I have worked on 30+ projects (academic and side projects).

- Data Bases II (CSEN 604)[4]

In this course, we studied the internal structure of a database engine including but not limited to query processing, logging and file organization.

- The Data Bases II (CSEN 604) PROJECT

In the project, we were required to implement a basic CSV driven DBMS system using Java.

- Internship done at Xlab, Alexandria, Egypt Summer 2016[5]

I implemented a backend API using Java (I know not the most suitable language for the task but this is what was required) that communicated through an ORM with PostgreSQL and when I was designing the database schema I only dreamt of this feature but was ***shut down*** *by my mentors. This is why I am particularly excited about this idea.*

Contact

Mail:	markm.rofail@gmail.com
Phone:	+201140388266
IM:	<google hangouts> markm.rofail@gmail.com <skype IM> markm.rofail <whatsapp> +201140388266 <facebook> https://www.facebook.com/MarkMRofail

References

[1]https://www.postgresql.org/message-id/16787.1351053391@sss.pgh.pa.u
[2]https://www.postgresql.org/message-id/28389.1351094795@sss.pgh.pa.us
[3]http://www.guc.edu.eg/en/academic_programs/course_catalog/course_details.aspx?courseId=51
[4]http://www.guc.edu.eg/en/academic_programs/course_catalog/course_details.aspx?courseId=59
[5]http://software.xlab-group.com/
[6]https://drive.google.com/open?id=0B_wjTOEciI-Jc2c4RGlzODlCTU0
[7]https://hashrocket.com/blog/posts/exploring-postgres-gin-index (I know this is a weak reference, I could not find a better one)