Easy Databases On Maemo

Part 1: Stop Writing Code
Part 2: Glom
Part 3: Handheld Glom
Part 1: Stop Writing Code

Developers are incredibly ineffective
Developers Are Difficult

- Argue about editors
- Argue about version control
- Argue about programming languages
- Argue about code style
- Argue about toolkits
- Argue about build tools
- Nice to have: Implement features
The Best Is Not Enough

- People say the best developers are 10 x more productive than other developers
- That's true
- That's still 1000 x not good enough
Please

Stop The Developers

Glom
Code

- Code crashes
- Code works but you don't know why
- Code does something but you don't know what
- Code stops working if you look at it.
100 Ways To Be Slightly Better

- Development methodologies
- Specifications
- IDEs
- New programming languages
- Regression Testing

After everything, code is still mostly awful.
Please

Stop Writing Code
Estimates Are Always Wrong

- Always 2 weeks until it's finished. Translation: "We have no idea".
- First month: Version 0.5, 0.6, 0.7, 0.8, 0.9
- Last two years: 0.91, 0.92 ... 0.9991 ... 0.9999, 2.0
- Requirements Always Change
Please

Stop Planning

Show useful results early. Move on.
Part 2: Glom

- Design database systems
- UI and structure together
- Self-contained in one folder
- For normal users
- No code. No SQL. (You can if you must)
Features (1)

- Simple field types
- Automatic layout
- Find
- Relationships:
  - Related records
  - Field lookups
  - Related fields
  - Drop-down choices
Desktop Uses

Anything that people would (ab)use Excel for.

- Order Taking and Invoicing
- Project Management
- Membership Databases
- Bookings, Reservations
- Film Production Management
Handheld Uses

**Jobs without computers:** People who work on their feet.

- Restaurant Orders
- Deliveries
- Stock Taking
- Warehouse Order Picking
- Garage Billing
- School Attendance
Layout: List

- Shows many records:

![Music Collection Table]

**Table: Artists**

<table>
<thead>
<tr>
<th>Artist ID</th>
<th>Name</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Bruce Springsteen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Madonna</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Prince</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Curtis Mayfield</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• Show more details for one record.
• Identify records based on a field value.
Using Relationships

- View related records
- View a related field
- Lookup a value from a related field
- Show list of possible values from a related record
No need for SQL

• This is pain:
  SELECT "invoices"."invoice_id", "invoices"."date", "invoices"."contact_id", "relationship_contacts"."name_full", "invoices"."price_total", "invoices"."vat_total", "invoices"."price_total_with_vat", "invoices"."comment", "invoices"."invoice_id" FROM "invoices" LEFT OUTER JOIN "contacts" AS "relationship_contacts" ON ("invoices"."contact_id" = "relationship_contacts"."contact_id") WHERE "invoices"."invoice_id" = 0

• Glom does it for you.
Extend via Python

- Define **Field calculations** using Python. To calculate a value based on other fields.

- Add a **button** that runs a Python script. To do something based on the field values, Use SQL if you like, via the provided pygda connection.
Extend via SQL

- Glom uses PostgreSQL or SQLite.
- So you can access the data directly, without even using Glom.
- Just don't use unusual features that Glom doesn't expect.
Extend via XML

- .glom files are XML.
- The format is obvious and documented.
- Libglom helps you to read the XML. Qlom (Qt) uses it. We have a Java JSP test too.
Part 3: Handheld Glom

- Client-only: Just viewing and Editing of data
- TouchSelector instead of TreeView
- PickerButton+TouchSelector instead of ComboBox
- Hide some labels
- Use single-columned layouts by default
- Use Hildon Buttons, Entries, Windows
- ifdeftastic
Simpler UI

List:

0  Bruce Springsteen
1  Madonna
2  Prince
3  Curtis Mayfield
4  Louis Armstrong
Simpler UI

Details:

Album ID: 3
Name: Sign 'O' The Times
Artist ID: 5
Name: fdf
Publisher ID: 1
Name: Warner Bros
Openismus GmbH

We Create Quality
We Fight Entropy
Easy Databases On Maemo

Part 1: Stop Writing Code
Part 2: Glom
Part 3: Handheld Glom
Part 1: Stop Writing Code

Developers are incredibly ineffective

Glom helps you to quickly write database-based applications..

Developing software is insanely difficult. But for these types of applications you often have a good idea of what it should do, and you don't want a mass of code between you and finding out if you are right about what it should do.

You need to quickly show a UI to the user and get some feedback. Then you need to quickly adapt the system and show it again before the user has forgotten what they wanted or thought of something extra.

Developers get in the way of this.
Developers have the strangest priorities.

Any 2 developers will disagree. If a developer has nobody to disagree with then he'll disagree with his manager instead.

Developers are obsessed with the little things they can understand. They don't mention the vague things they can't have an opinion about. Users are vague. Customers are vague. Deadlines are vague. Non-Developers cannot decide which developer is correct. Don't try.

Developers say you must do things you don't understand. You have no way of knowing if it's really worth the time. You have a 50% chance of getting it wrong. The
The Best Is Not Enough

- People say the best developers are 10 x more productive than other developers
- That's true
- That's still 1000 x not good enough
Please

Stop The Developers

Developers create far more problems than they solve. Be in the happy user business, not the coding business.
More code means more problems.

Developers say: "It's only 1000 lines of code". It will have at least 1000 bugs.
Developers say "You just need to …"
Followed by a huge amount of incomprehensible stuff. This is not understanding. This is pretending to understand. It's voodoo. Don't lie to yourself.
Glom

Please

Stop Writing Code

I don't mean it personally. You're all nice people. You're from good families. You've just got mixed up with some bad people and developed some nasty habits.

OK, so some of you are creating new platforms or applications with unique Uis. Most developers are just like visual basic developers – I think we can agree that they should stop. The world doesn't need their code.
Openismus uses the Scotty technique. We overestimate wildly so the customer is insanely happy when you finish early. Because developers never finish early.

In summary, estimates are science fiction.
Stop Planning

Show useful results early. Move on.

Stop lying to yourselves.

Refactoring:
- New requirements can require massive rewrites. You don't know how disruptive until you start.
- More code means more code to change, test, fix.
Part 2: Glom

- Design database systems
- UI and structure together
- Self-contained in one folder
- For normal users
- No code. No SQL. (You can if you must)
Features (1)

- Simple field types
- Automatic layout
- Find
- Relationships:
  - Related records
  - Field lookups
  - Related fields
  - Drop-down choices

Also reports, users/groups.
**Desktop Uses**

Anything that people would (ab)use Excel for.

- Order Taking and Invoicing
- Project Management
- Membership Databases
- Bookings, Reservations
- Film Production Management

So what might you use Glom for?

Also roughly equivalent to Access, but far less painful. Very similar to FileMaker Pro.
Handheld Uses

Jobs without computers: People who work on their feet.

- Restaurant Orders
- Deliveries
- Stock Taking
- Warehouse Order Picking
- Garage Billing
- School Attendance
Layout: List

- Shows many records:
• Show more details for one record.
## Relationships

- Identify records based on a field value.

<table>
<thead>
<tr>
<th>Name</th>
<th>Title</th>
<th>From Field</th>
<th>Table</th>
<th>To Field</th>
<th>Automatic Creation</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoice_lines</td>
<td>invoice_lines</td>
<td>invoice_lines</td>
<td>invoice_lines</td>
<td>invoice_id</td>
<td>false</td>
</tr>
<tr>
<td>contacts</td>
<td>contact</td>
<td>contact_id</td>
<td>contacts</td>
<td>contact_id</td>
<td>false</td>
</tr>
</tbody>
</table>

![Relationships Diagram](image-url)
Using Relationships

- View related records
- View a related field
- Lookup a value from a related field
- Show list of possible values from a related record
No need for SQL

- This is pain:
  SELECT "invoices"."invoice_id", "invoices"."date", "invoices"."contact_id", "relationship_contacts"."name_full", "invoices"."price_total", "invoices"."vat_total", "invoices"."price_total_with_vat", "invoices"."comment", "invoices"."invoice_id" FROM "invoices" LEFT OUTER JOIN "contacts" AS "relationship_contacts" ON ("invoices"."contact_id" = "relationship_contacts"."contact_id") WHERE "invoices"."invoice_id" = 0

- Glom does it for you.
Extend via Python

- Define **Field calculations** using Python. To calculate a value based on other fields.

- Add a **button** that runs a Python script. To do something based on the field values, Use SQL if you like, via the provided pygda connection.
Extend via SQL

- Glom uses PostgreSQL or SQLite.
- So you can access the data directly, without even using Glom.
- Just don't use unusual features that Glom doesn't expect.
Extend via XML

- .glom files are XML.
- The format is obvious and documented.
- Libglom helps you to read the XML. Qlom (Qt) uses it. We have a Java JSP test too.
Part 3: Handheld Glom

- Client-only: Just viewing and Editing of data
- TouchSelector instead of TreeView
- PickerButton+TouchSelector instead of ComboBox
- Hide some labels
- Use single-columned layouts by default
- Use Hildon Buttons, Entries, Windows
- ifdeftastic
Simpler UI

List:

0. Bruce Springsteen
1. Madonna
2. Prince
3. Curtis Mayfield
4. Louis Armstrong
Simpler UI

Details:

Album ID: β
Name: Sign ‘O’ The Times
Artist ID: 5 Open Find
Name: fdf
Publisher ID: 1 Open Find
Name: Warner Bros
Openismus GmbH

We Create Quality
We Fight Entropy