

MonetDB & Python

Gijs Molenaar, gijs@pythonic.nl

September 24, 2009

Outline

- 1 Introduction
- 2 MonetDB
- 3 Python API
- 4 Final

About me

- Gijs Molenaar
- student MSc A.I. at UvA
- Working freelance for CWI / MonetDB

Outline

- 1 Introduction
- 2 MonetDB
 - About MonetDB
 - What is it good for
 - Unique properties
 - BAT
 - XML
 - Latency bottleneck
 - API'S
- 3 Python API
- 4 Final

About MonetDB

- DBMS, developed by CWI
- open source, BSD license
- Platform for scientific research
- Not only SQL
- Available at www.monetdb.nl
- Deb and RPM packages available



What is it good for

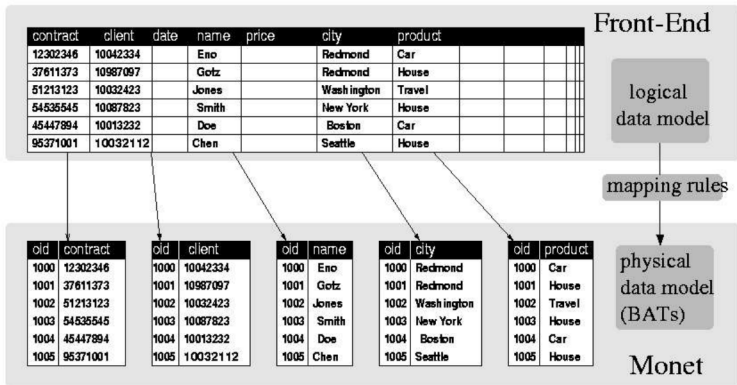
- Big databases (Skyserver, 6TB)
- Query intensive apps (ro)
- Research
- Complex data models, beyond SQL
- Sometimes outperformes other DB's by x100 (TPC-H)

<http://www.monetdb.nl/projects/monetdb//SQL/Benchmark/TPCH/>

Unique properties

- Column based storage database
- BAT - Binary Association Table
- Flexible kernel, easy extendable
- SQL frontend
- But Not limited to SQL
- For example Xquery, XML database

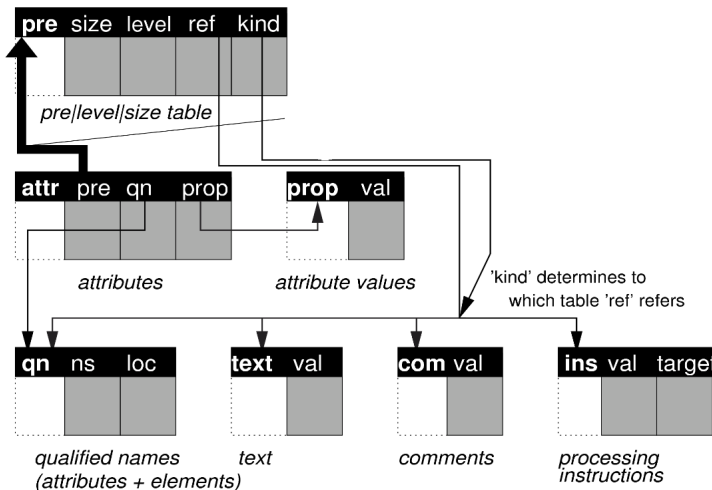
Column based storage database (BAT)



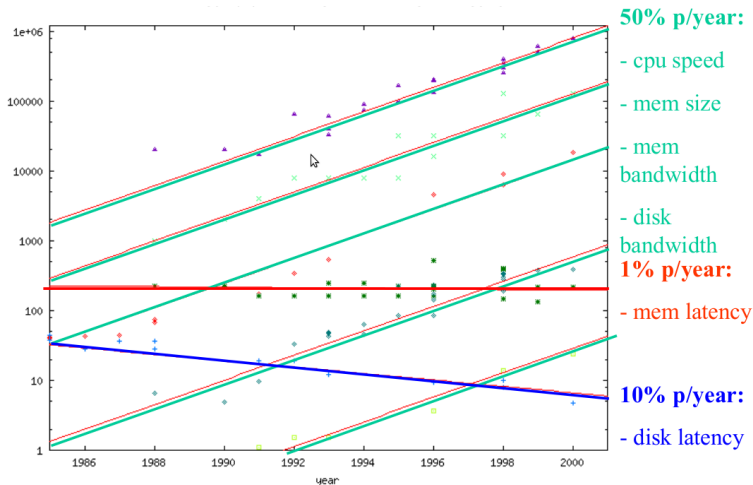
XML

- MonetDB-Xquery
- Store XML in DB
- Structure is translated to BAT
- BUT: MonetDB4 only
- python API not really good
- out of scope

XML 2



Bandwidth bottleneck



API'S

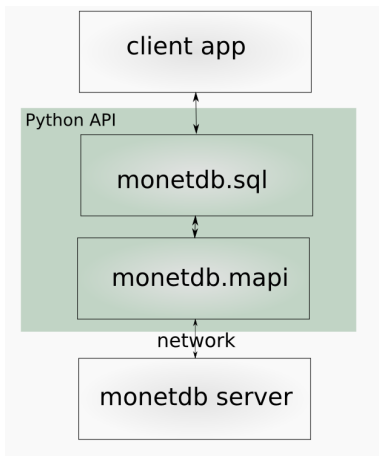
- Ruby (native) - by Gabriele Modena
- PHP - by Gabriele
- python
- Perl - Nobody wants to get near
- C
- Java (native)

Outline

- 1 Introduction
- 2 MonetDB
- 3 Python API**
 - Requirements
 - Design
 - Testing
 - Python 2 & 3
 - example
 - Conclusion
- 4 Final

- Total rewrite of current API
- Native implementation
- Python DBAPI 2.0
- Better documentation
- Support all Pythons
- Other good stuff

Design



python.mapi

- MonetDB API, wire protocol
- protocol is text (not binary)
- Basic functions:
 - Set up connection
 - Execute commands
 - Parse response
 - chop queries and responses in blocks

python.sql

- Implementation of all DBAPI 2.0 functionality
- Bridge between DBAPI functions and mapi

Testing

- Good idea, never did it
- Python DBAPI2 tests already written
- Borrowed tests from MySQL python API

Guido said

- Start with excellent test coverage.
- Run python2.6 with -3. Run tests, fix warnings.
- Run 2to3.py
- Run tests, fix remaining errors
- Don't try programming mixed 2 & 3 code

But

- Maintaining 2 trees is more work
- Code duplication
- Not so many (big) problems
- Not porting, rewriting
- Decided to create a 2/3 hybrid

Problems

- Print
- Unicode
- New try except syntax
- Python2.5 != 2.6 != 3.0
- New IO layer (bytes, strings)
- DBAPI 2.0 not python 3 compatible

New IO layer

- Used in mapi layer
- Strict separation of bytes and strings
- Socket only accepts bytes
- Too many changes
- 2 versions, mapi2.py and mapi3.py
- +/- 180 lines of code

Ugly

Version check

```
major = sys.version_info[0]
if major == 3:
    ..
else:
    ..
```

- Used only 2 times
- In unittest
- In monetsql, import mapi2 or mapi3

print

- Don't use print
- Use logging

logging debug

```
logging.basicConfig(level=logging.DEBUG)
logger = logging.getLogger('monetdb')
```

Unicode

- MonetDB supports UTF-8
- If user input is only ASCII or UTF-8, no problem
- Only in Unittests a problem
- u'python' not working in Python3, syntax error

solution

```
2: unicode('\Nlatin small letter a with acute')
```

```
3: '\N{latin small letter a with acute}'
```

New Try Except syntax

- 2: `except socket.error, error`
- 3: `except socket.error as error`
- Only used in mapi layer
- Solved `mapi2.py` and `mapi3.py`

overview

- 2 separated files for low level core (mapi2, mapi3)
- Only 2 times ugly version check

example

```
>import monetdb.sql
>connection = monetdb.sql.connect()
>cursor = connection.cursor()
>cursor.execute('SELECT * FROM tables')
26
>cursor.fetchone()
1062, 'schemas', 1061, None, 0, True, 0, 0
```

Conclusion

- Hybrid model works quite good (in this case)
- Because of setup and simplicity
- Minimized code duplication
- Not advised for all projects

Outline

- 1 Introduction
- 2 MonetDB
- 3 Python API
- 4 Final**
 - Future Work

Future work

- Bug fixes
- Port Xquery API (?)
- Currently rewriting C API (mapilite)
- Unittests in python using ctypes
- Problem with structs

Something else

- Setting up pythonic.nl
- Non-profit amateur webhosting
- Python django focus
- Small group
- Maximum flexibility
- Need users!

pythonic.nl

Questions?