

# GT.M

NoSQL in core banking systems

@zmerta

NoSQL in core banking systems?

**WTF?**

The story begins as usual...

A long time ago in a galaxy far,  
far away....

A long time ago in a galaxy far,  
far away....

**MUMPS**

A long time ago in a galaxy far,  
far away....

**MUMPS**

Welcome to the 60s!

# MUMPS

Massachusetts General Hospital **U**tility **M**ulti-**P**rogramming **S**ystem  
or simply **M**

## Features

- ANSI and ISO standardized
- Procedural language combined with a database engine
- ACID transactions
- Optimized for low-memory computers
- Commercial implementations: GT.M, Caché

# MUMPS

## Syntax

- Weird language constructions
- All commands could be shorten to one or two letters
- Allows multiple commands on a line
- Beware of white spaces - some are significant



# MUMPS

```
%DTC ; SF/XAK - DATE/TIME OPERATIONS ;1/16/92 11:36 AM
;;19.0;VA FileMan;;Jul 14, 1992
D I 'X1!'X2 S X="" Q
S X=X1 D H S X1=%H,X=X2,X2=%Y+1 D H S X=X1-%H,%Y=%Y+1&X2
K %H,X1,X2 Q
;

C S X=X1 Q:'X D H S %H=%H+X2 D YMD S:$P(X1,".",2) X=X_"."_$P(X1,".",2)

K X1,X2 Q

S S %=#60/100+(%#3600\60)/100+(%\3600)/100 Q
;

H I X<1410000 S %H=0,%Y=-1 Q
S %Y=$E(X,1,3),%M=$E(X,4,5),%D=$E(X,6,7)
S %T=$E(X_0,9,10)*60+$E(X_"000",11,12)*60+$E(X_"00000",13,14)

TOH S %H=%M>2&'(%Y#4)+$P("^31^59^90^120^151^181^212^243^273^304^334","^",%M)+%D
S %='%M!'%D,%Y=%Y-141,%H=%H+(%Y*365)+(Y\4)-(Y>59)+%,%Y=$S(%:-1,1:%H+4#7)
K %M,%D,% Q
```

**OMG!!!**

It's even worse than Perl

or Brainfuck...

or Whitespace...

:-)

It's not so bad...

MUMPS could be readable...

# MUMPS

```
HELLO ; Hello application
      ;
      FOR I=1:1:10 DO
      . IF I#2=0 DO
      .. DO PRINT^HELLO("Peter")
      . ELSE DO
      .. DO PRINT^HELLO("Paul")
      QUIT
      ;
PRINT(NAME)
      WRITE "Hello, ",NAME,!
      QUIT
```

Nobody wants to write  
large applications in MUMPS

Think about it as a language for  
writing stored procedures  
and triggers...

But there is one aspect which is still  
ahead of time...



# Global Persistent Variables aka Globals

# MUMPS - Globals

## Features

- Hierarchical key-value data store (persistent arrays)
- B-trees abstraction
- Keys are ordered
- Schema-less

# MUMPS - Globals

```
^ACCOUNTS ---
|
|----- 1 -----
|               |
|               |----- name --- Petr Novák
|               |
|               |----- email --- petr.novak@mail.cz
|
|----- 2 -----
|               |
|               |----- name --- Martina Pavlíčková
|               |
|               |----- email --- martina.pavlickova@mail.cz
```

---

```
^ACCOUNTS(1,"name") = "Petr Novák"
|                   |
|                   |----- subscript
|                   |
|                   |----- node
|
|----- global
```

Oh wait, isn't it...

NoSQL?

NoSQL?

50 years ago?

Even better...

Globals are very flexible...



# MUMPS - Globals

Can be used to model:

- Relational databases
- Key value databases (Redis, memcached, ...)
- Column/tabular databases (BigTable, Cassandra, ...)
- Document databases (MongoDB, CouchDB, ...)
- Graph databases (Neo4j, ...)

Later in the same galaxy...

Later in the same galaxy...

**GT.M**

# GT.M

Greystone Technology M

## Features

- Database engine and application platform
- MUMPS implementation
- The heart of FIS Profile™ (core banking system)
- Used in large financial and healthcare institutions
- Open sourced in 2000 (GPLv2, AGPL)
- Multiplatform (AIX, Linux, OpenVMS, Solaris, z/OS...)
- Commercial support

# GT.M

## Performance

- High performance
- Small footprint
- Ultra fast ACID transactions (optimistic concurrency)
- Software transactional memory
- In-memory like performance with on-disk like integrity
- Sophisticated caching

# GT.M

## Security

- Uses underlying operating system
- Processes run as user level processes
- No database daemon
- Journaling
- Database encryption
- Plugin architecture for encryption algorithms

# GT.M

## Scalability & continuity of business

- Logical multi site operation
- One instance streams to up to 16 replicas
- 2 replication modes:
  - BC replication
  - SI replication

# GT.M

## Deployment

- Internet superserver (inetd, xinetd)
- EWD (M/Gateway)
- MTM (Profile)



# GT.M

## Limits

- One Global must fit in one database file  
(no limit on the number of database files)
- Database file is divided to 992 mi blocks  
(variable block sizes - multiple of 512 bytes, with a maximum size of 65,024)

# GT.M

Proven in business-critical environments

World's largest banks

- ING DIRECT
- Capital One 360 (USA)
- Barclays Direct (GB)

Healthcare

- WorldVistA (Electronic Health Record system)

# NoSQL - Advantages

## Five Advantages

1. Elastic scaling
2. Big data
3. Goodbye DBAs
4. Economics
5. Flexible data models

# NoSQL

## Five Advantages

1. Elastic scaling  
Support for scaling out across multiple servers
2. Big data
3. Goodbye DBAs
4. Economics
5. Flexible data models

# NoSQL

## Five Advantages

1. Elastic scaling
2. Big data  
Designed to support huge amount of data
3. Goodbye DBAs
4. Economics
5. Flexible data models

# NoSQL

## Five Advantages

1. Elastic scaling
2. Big data
3. Goodbye DBAs  
Some systems have been running unattended for decades
4. Economics
5. Flexible data models

# NoSQL

## Five Advantages

1. Elastic scaling
2. Big data
3. Goodbye DBAs
4. Economics  
Runs on low-cost commodity hardware
5. Flexible data models

# NoSQL

## Five Advantages

1. Elastic scaling
2. Big data
3. Goodbye DBAs
4. Economics
5. Flexible data models  
NoSQL or relational databases emulation



# NoSQL

## Five challenges

1. Maturity
2. Support
3. Analytics and business intelligence
4. Administration
5. Expertise

# NoSQL

## Five challenges

1. Maturity  
50. years in business-critical environments
2. Support
3. Analytics and business intelligence
4. Administration
5. Expertise

# NoSQL

## Five challenges

1. Maturity
2. Support  
Commercial support available.
3. Analytics and business intelligence
4. Administration
5. Expertise

# NoSQL

## Five challenges

1. Maturity
2. Support
3. Analytics and business intelligence  
DeepSee - product for realtime analysis  
(InterSystems / Caché)
4. Administration
5. Expertise

# NoSQL

## Five challenges

1. Maturity
2. Support
3. Analytics and business intelligence
4. Administration  
Easy to install and to maintain.
5. Expertise

# NoSQL

## Five challenges

1. Maturity
2. Support
3. Analytics and business intelligence
4. Administration
5. Expertise  
Small number of skilled professionals :-)

So why we reinventing the wheel  
over and over again?