

TUGAS KAPITA SELEKTA



Disusun Oleh :

Edi Sukrisno

09011181320043

Fakultas Ilmu Komputer

Sistem Komputer

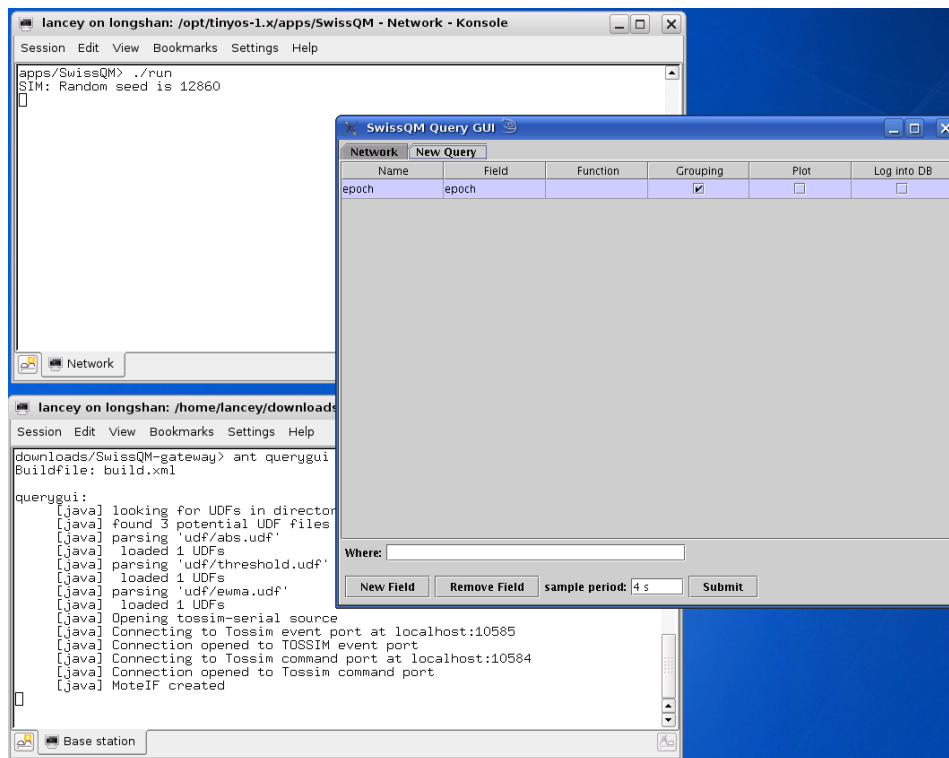
2016

Analisa IoT Middleware

SwissQM

SwissQM adalah antarmuka pemrograman aplikasi jaringan *sensor network*. *Queries* (permintaan untuk mengumpulkan data dari jaringan nirkabel sensor) dapat disusun oleh pengguna di antarmuka grafis dari SwissQM. Permintaan kemudian dikompilasi ke dalam program yang dikirim ke jaringan sensor dimana mereka dieksekusi.

Ketika dijalankan maka tiga jendela akan terbuka pada desktop, seperti yang ditunjukkan dalam gambar 1. Jangan ditutup salah satunya, Jika menutup salah satu jendela akan terjadi kesalahan program. Dalam gambar 1, jendela *SwissQM Query GUI* adalah antarmuka pengguna grafis (GUI) dari SwissQM, mana query disusun dan disampaikan. *Network window* menunjukkan hasil simulasi, termasuk sampel data pada setiap node dan data transmisi peristiwa dari node 0 pada stasiun base. *windows Stasiun Base* menunjukkan permintaan eksekusi informasi dan data yang dikumpulkan di stasiun base.



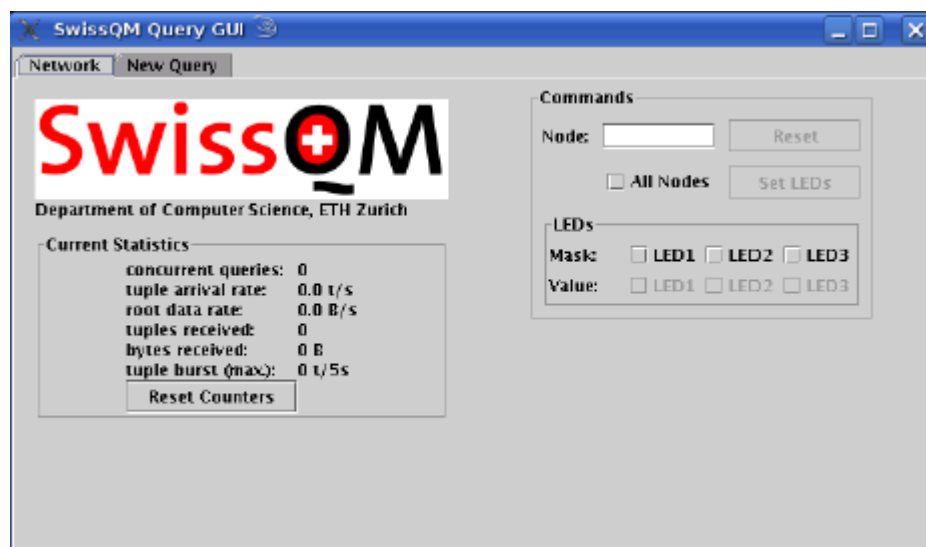
Gambar 1. Tampilan awal *SwissQM*

1. Menggunakan *SwissQM Query GUI*

SwissQM Query GUI digunakan untuk menulis *Query*, mengirim mereka ke jaringan, dan plot data yang dikumpulkan pada *base station*. GUI terdiri dari tiga tab: tab *Network*, tab *Query* baru, dan tab *Plot*. Tab *Plot* muncul hanya setelah query dengan persyaratan plot disampaikan.

2. *Network Tab*

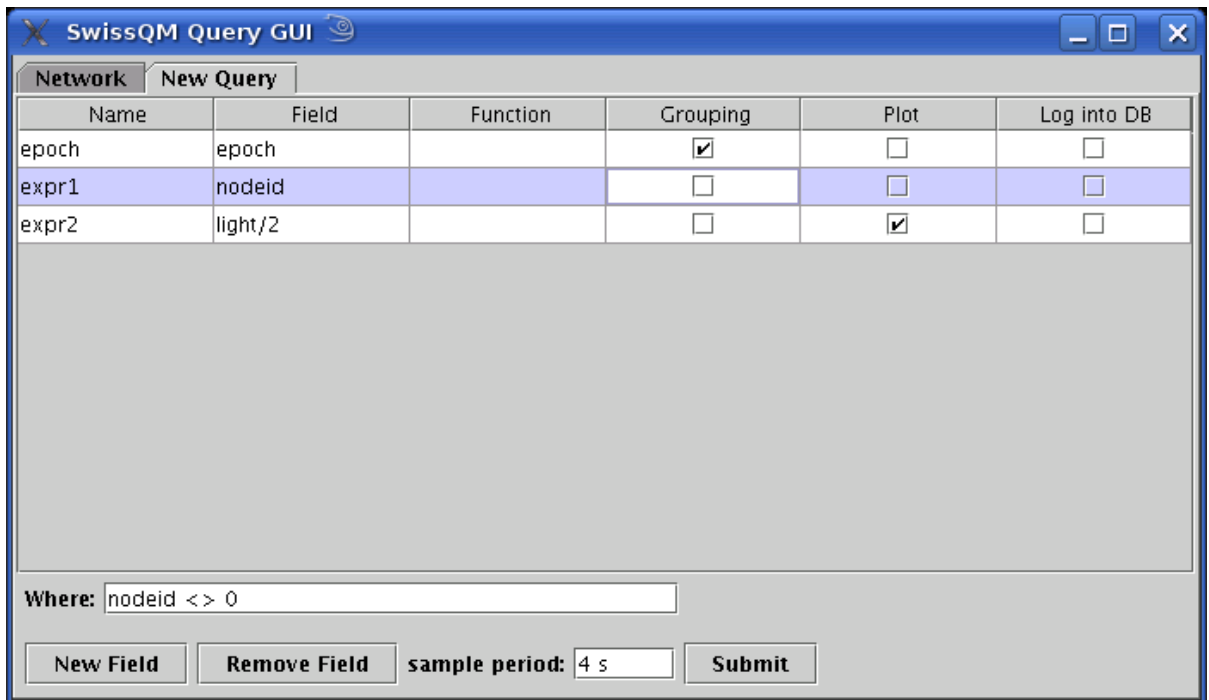
Pada gambar 2 menunjukkan *Tab Network*. Kolom kiri menunjukkan data statistik komunikasi antara *base station* dan *sensor network*. Kolom kanan menunjukkan perintah yang dapat dikirim ke jaringan. Anda tidak perlu menggunakan kolom kanan dalam tes ini, sehingga Anda bisa mengabaikannya.



Gambar 2. SwissQM GUI

3. *New Query Tab*

Queries terdiri dalam *Query Tab* Baru, seperti yang ditunjukkan pada Gambar 3. zaman lapangan pertama selalu hadir dan tidak dapat diubah . Hal ini digunakan sebagai cap kasar dari data. Ini menghitung berapa banyak waktu pengambilan sampel telah berlalu sejak awal *query*. Bidang *Query* ditambahkan dengan mengklik tombol *New Field*. Sebuah baris baru ditambahkan ke *table* ketika *fields* baru ditambahkan. Anda dapat mengedit atribut (kolom) dari ladang untuk menulis *query*.



Gambar 3. Query for example application.

Sebuah *query* dapat berisi beberapa bidang dan pertanyaan baru dapat diserahkan sementara yang lain terus berjalan. Penjelasan dari atribut *fields* sebagai berikut:

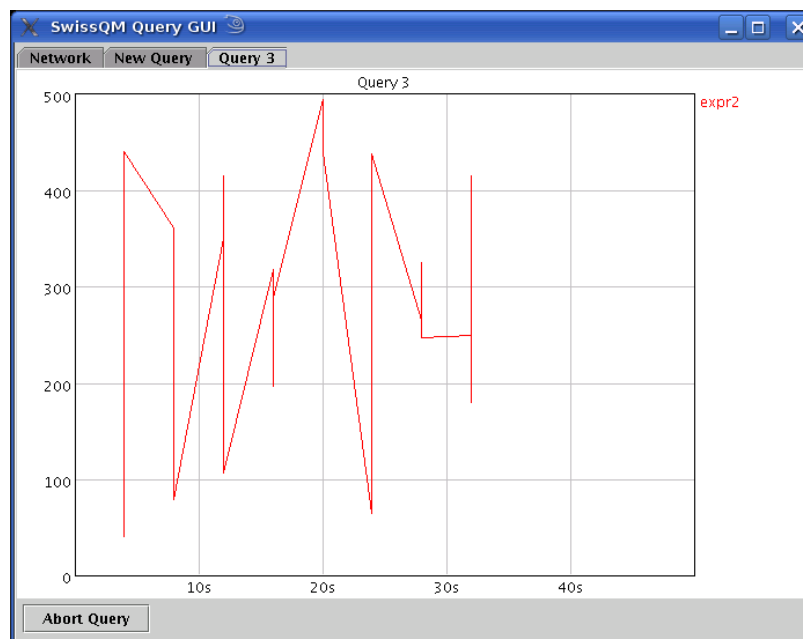
- *Name: Identifier* (tidak ada spasi) dari *fields* itu. Pengenal ini ditampilkan dalam legenda plot hasil. Teks dapat diedit secara langsung.
- *Field: Field expression* menggambarkan data yang akan dikumpulkan. Mengklik pada kotak teks bidang akan menunjukkan daftar drop-down yang berisi data sensor pada sebuah node. Bidang teks kotak dapat diedit untuk menulis ekspresi. Ekspresi adalah omposed Operand, operator dan kurung. Operator adalah salah satu dari + (Tambahkan), - (kurangi), * (kalikan), dan / (membagi). Operan dapat item dari daftar drop-down atau integer. Untuk contoh, (cahaya + 100) * 2 adalah sebuah ungkapan yang dapat dimasukkan dalam kotak teks lapangan.
- *Function*: Mengklik pada bidang teks fungsi akan menunjukkan daftar drop-down yang mengandung semua fungsi built-in. Deskripsi fungsi ini adalah dalam tabel 1. Fungsi yang dipilih diterapkan ke bidang kolom. Bila Anda memilih fungsi dengan mengkliknya, itu akan dimasukkan ke bidang kolom. Agregasi fungsi seperti min dan jumlah berlaku untuk nilai-nilai bidang dari node dalam kelompok yang sama. Sensor node dikelompokkan dengan

pengelompokan bendera, yang akan dijelaskan kemudian. Fungsi kotak teks dapat diedit secara langsung.

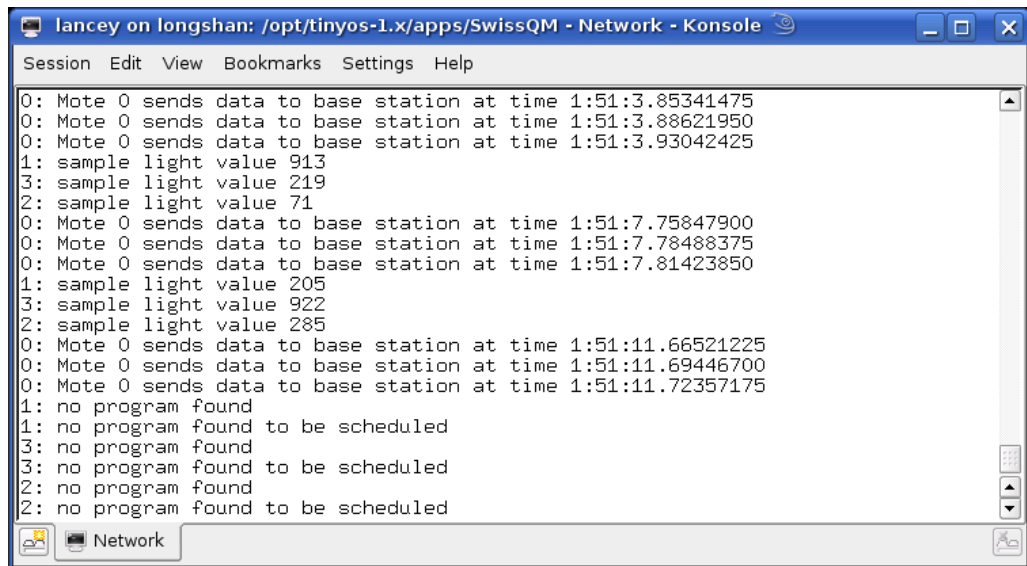
Function name	Arguments	Returned data
variance	<i>field</i>	variance of <i>field</i>
ewma	<i>field</i>	exponentially weighted moving-average of <i>field</i>
min	<i>field</i>	the minimum of <i>field</i>
threshold	<i>field, threshold</i>	1 if the value of <i>field</i> is larger than <i>threshold</i> , otherwise 0
stddev	<i>field</i>	the standard deviation of <i>field</i>
avg	<i>field</i>	the average value of <i>field</i>
count		the number of nodes in a group
sum	<i>field</i>	the sum of values of <i>field</i>
mod	<i>field, integer</i>	remainder of <i>field</i> divided by integer
max	<i>field</i>	the maximum of <i>field</i>
abs	<i>field</i>	the absolute value of <i>field</i>

Tabel 1. Built-in fungsi

- Grouping: Menetapkan bendera ini jika Anda ingin membagi node menjadi kelompok menurut data di kolom bidang yang sesuai. Node dengan nilai yang sama dari bidang akan dikelompokkan bersama. Pengelompokan bidang tidak dapat dipetakan. Jika tidak ada bendera pengelompokan terletak di query, tetapi menggunakan fungsi agregasi, maka operasi agregasi berlaku untuk semua node dalam jaringan.
- Plot: Nilai dari field akan diplot jika ia akan diperiksa.
- Log into DB: nilai dari field akan mulai bisa login ke dalam database jika ia akan diperiksa.

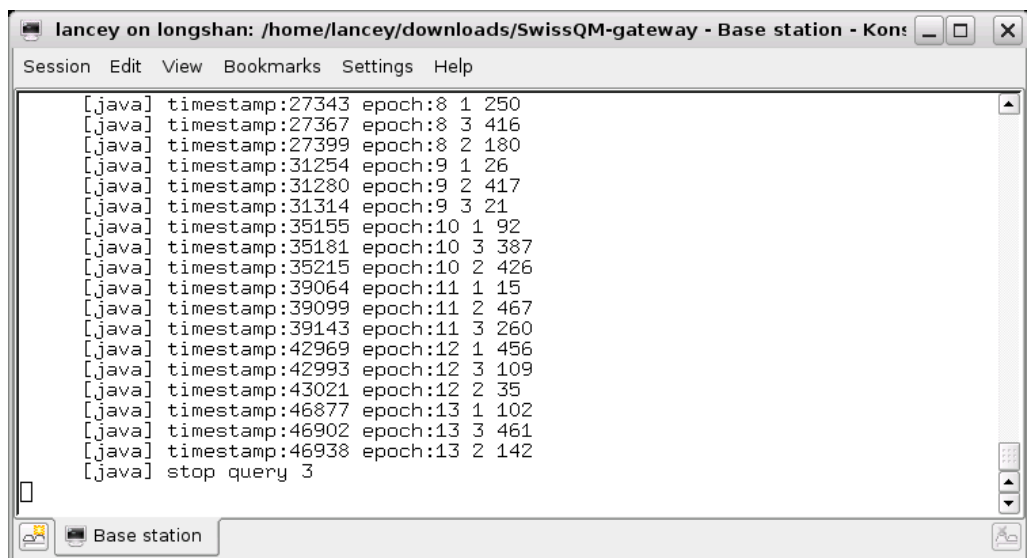


Gambar 4. Contoh plot dari *SwissQM*



```
lancey on longshan: /opt/tinyos-1.x/apps/SwissQM - Network - Konsole
Session Edit View Bookmarks Settings Help
0: Mote 0 sends data to base station at time 1:51:3.85341475
0: Mote 0 sends data to base station at time 1:51:3.88621950
0: Mote 0 sends data to base station at time 1:51:3.93042425
1: sample light value 913
3: sample light value 219
2: sample light value 71
0: Mote 0 sends data to base station at time 1:51:7.75847900
0: Mote 0 sends data to base station at time 1:51:7.78488375
0: Mote 0 sends data to base station at time 1:51:7.81423850
1: sample light value 205
3: sample light value 922
2: sample light value 285
0: Mote 0 sends data to base station at time 1:51:11.66521225
0: Mote 0 sends data to base station at time 1:51:11.69446700
0: Mote 0 sends data to base station at time 1:51:11.72357175
1: no program found
1: no program found to be scheduled
3: no program found
3: no program found to be scheduled
2: no program found
2: no program found to be scheduled
```

Gambar 5. Contoh hasil dari *network window*



```
lancey on longshan: /home/lancey/downloads/SwissQM-gateway - Base station - Konsole
Session Edit View Bookmarks Settings Help
[java] timestamp:27343 epoch:8 1 250
[java] timestamp:27367 epoch:8 3 416
[java] timestamp:27399 epoch:8 2 180
[java] timestamp:31254 epoch:9 1 26
[java] timestamp:31280 epoch:9 2 417
[java] timestamp:31314 epoch:9 3 21
[java] timestamp:35155 epoch:10 1 92
[java] timestamp:35181 epoch:10 3 387
[java] timestamp:35215 epoch:10 2 426
[java] timestamp:39064 epoch:11 1 15
[java] timestamp:39099 epoch:11 2 467
[java] timestamp:39143 epoch:11 3 260
[java] timestamp:42969 epoch:12 1 456
[java] timestamp:42993 epoch:12 3 109
[java] timestamp:43021 epoch:12 2 35
[java] timestamp:46877 epoch:13 1 102
[java] timestamp:46902 epoch:13 3 461
[java] timestamp:46938 epoch:13 2 142
[java] stop query 3
```

Gambar 5. Contoh hasil dari *Base station window*

Kesimpulan

- SwissQM: platform untuk mengembangkan sensor aplikasi jaringan, bukan jaringan sensor solusi sendiri
- Menawarkan baik didefinisikan dan extensible bytecode bahasa
- Dapat digunakan sebagai target kompilasi lebih tinggi pemrograman dan bahasa query (XQuery,SQL)

Referensi :

- [1] Middleware for Internet of Things: a Survey. M.A. Razzaque, Marija Milojevic-Jevric, Andrei Palade, Siobh´an Clarke. 2015.
- [2] R. Mueller, G. Alonso, and D. Kossmann, “Swissqm: Next generation data processing in sensor networks.” in CIDR, 2007, pp. 1–9.