Euroforth 2020 “Rome”

A radical alternative to the Windows registry

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

THE GOOD OLD REGISTRY

* One of the best bits of Windows

* Gradually made less and less useful over the years

* Essentially, a hierarchical database

* Amazingly, there is no really good alternative in Linux

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

We used it for:

a) Basic configuration data *A BIT OF THIS*
e.g. Where to find the database

b) System configuration data ***LOTS*** *OF THIS*
e.g. How many Kgs do you put in a washing machine?

c) Per PC configuration data *A BIT OF THIS*
e.g. On this PC, are you allowed access to this dialog?

d) Small amounts of persistent data *A BIT OF THIS*
e.g. Go back to the same report that you selected last time

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

HOW TO EDIT THE DATA?

a) Basic configuration data — *A BIT OF THIS*
Need to set this manually, before the program will run

b) System configuration data — ***LOTS*** *OF THIS*
Dialog boxes ***ESSENTIAL*** *for managing the complexity*

c) Per PC configuration data — *A BIT OF THIS*
SIMPLE, so could be set manually

d) Small amounts of persistent data — *A BIT OF THIS*
Never needs setting

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

REPLACE THE REGISTRY WITH WHAT?

a) Basic configuration data *Configuration file*	*A BIT OF THIS*
b) System configuration data *PROBLEM*	***LOTS*** *OF THIS*
c) Per PC configuration data *Configuration file*	*A BIT OF THIS*
d) Small amounts of persistent data *Configuration file*	*A BIT OF THIS*

A radical alternative to the Windows registry

Configuration files – easy

Lots of possibilities

Libconfig is our favourite

Typical FORTH interface to Libconfig

```
: DATABASECONF ( --- ) \ Database settings
  Z" Database" CONFKEY
\   <length>      <address>       <Name>                               <Default>          <action>
    HOST_NAME_MAX ZDBHOST           Z" IP_address_of_normal_database"  Z" 192.168.0.10"  CONF$
    HOST_NAME_MAX ZDBCENTRAL        Z" IP_address_of_central_database" Z" 192.168.0.100" CONF$
    MAX_DB_USER   DBUSER            Z" Database_user"                  Z" Nick"          CONF$
    MAX_DB_PASS   DBPASS            Z" Database_password"              Z$SECRET          CONF$
    MAX_DB_NAME   DBNAME            Z" Database_name"                  Z" Tracknet"      CONF$
                  ADDR DBPORT       Z" Database_port_number"           3306              CONFVAR
;
```

Euroforth 2020 “Rome”

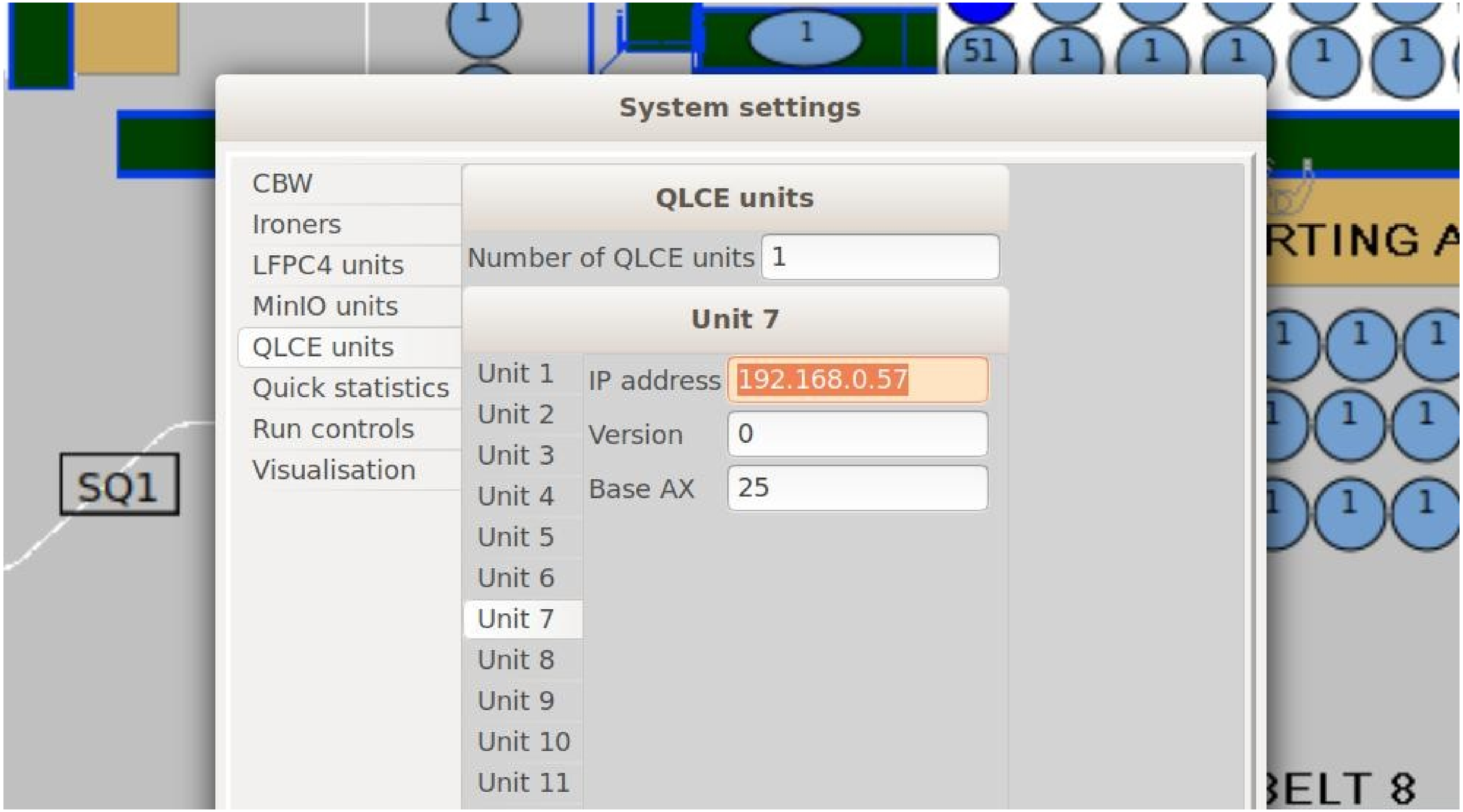
A radical alternative to the Windows registry

WHAT TO DO ABOUT SYSTEM CONFIGURATION DATA?

* Lots of it
* Needs to be easily editable
(But only by the configuration engineer)
* Needs to be structured
* Needs to be accessible from all devices on the control network
* Needs to be very flexible, and easily extensible

THIS SUGGESTS
a) A database table
b) A tabbed dialog box
c) The dialog box to be dynamically generated

A radical alternative to the Windows registry



QLCE units

Number of QLCE units: 1

Unit 7

IP address: 192.168.0.57

Version: 0

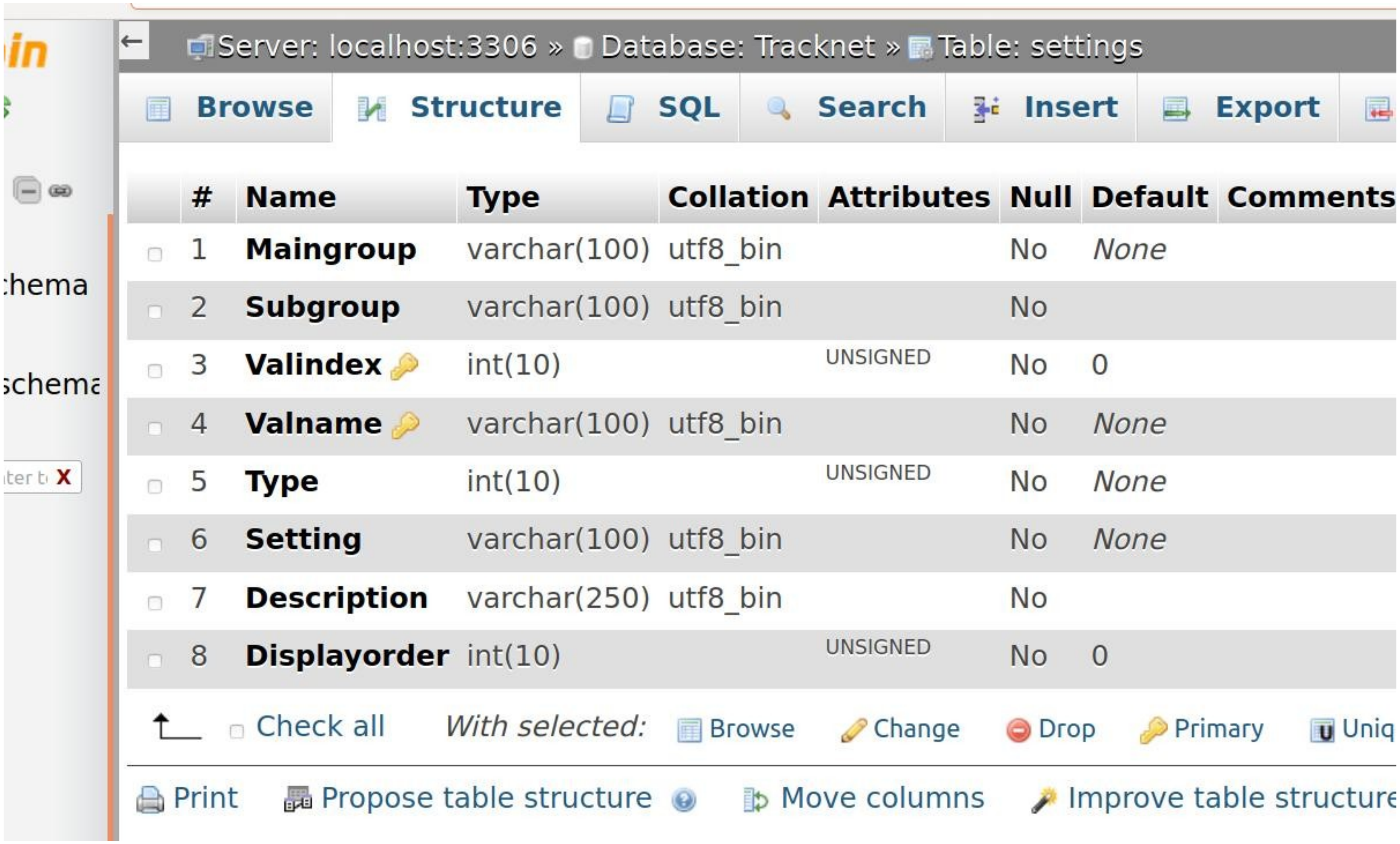
Base AX: 25

QLCE = Quad Loadcell to Ethernet

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

Database table columns



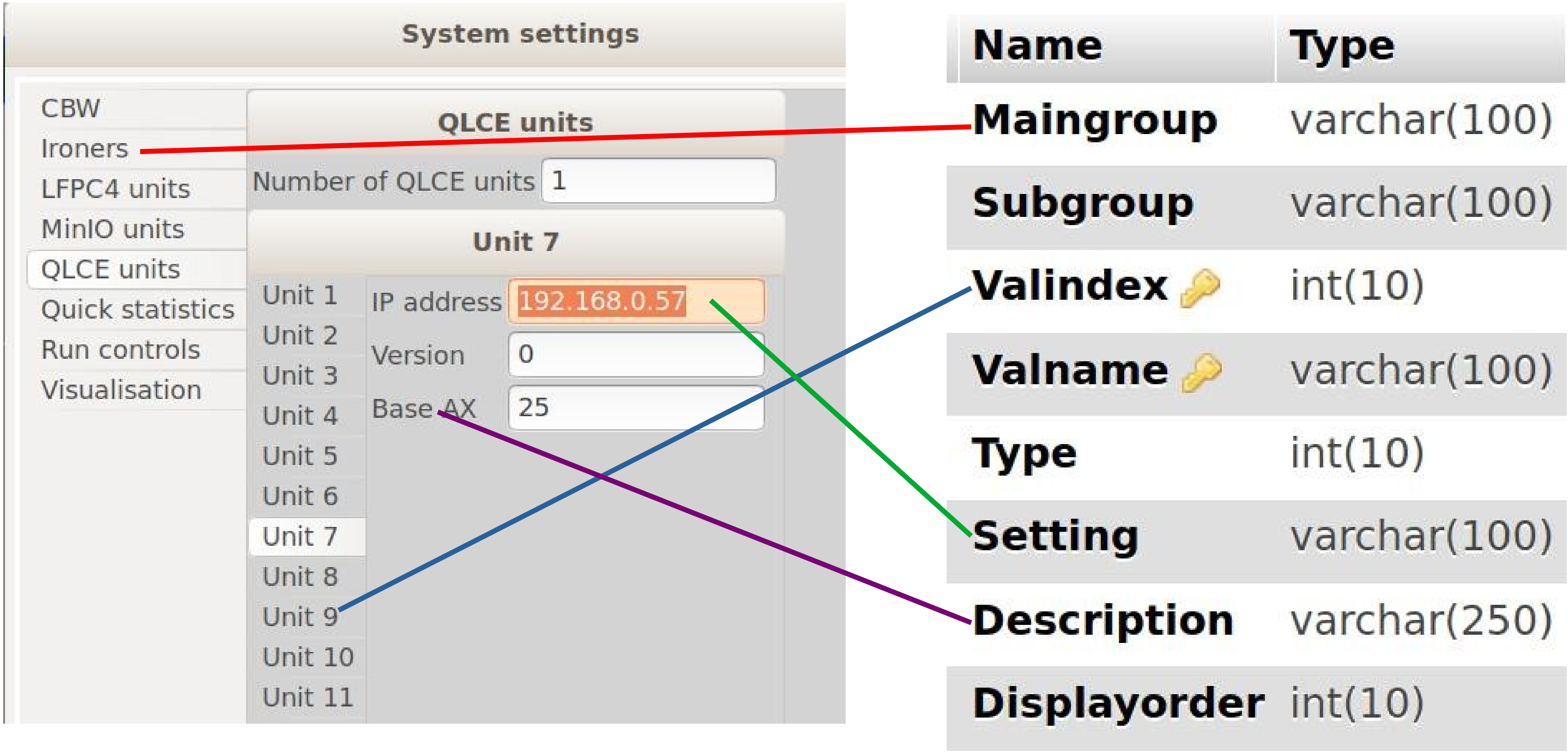
Server: localhost:3306 » Database: Tracknet » Table: settings
Browse
Structure
SQL
Search
Insert
Export
in
chema
schema
ter t X

	#	Name	Type	Collation	Attributes	Null	Default	Comments
□	1	**Maingroup**	varchar(100)	utf8_bin		No	*None*	
□	2	**Subgroup**	varchar(100)	utf8_bin		No		
□	3	**Valindex**	int(10)		UNSIGNED	No	0	
□	4	**Valname**	varchar(100)	utf8_bin		No	*None*	
□	5	**Type**	int(10)		UNSIGNED	No	*None*	
□	6	**Setting**	varchar(100)	utf8_bin		No	*None*	
□	7	**Description**	varchar(250)	utf8_bin		No		
□	8	**Displayorder**	int(10)		UNSIGNED	No	0	

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

Tying the DB table to the dialog



System settings

- CBW
- Ironers
- LFPC4 units
- MinIO units
- QLCE units
- Quick statistics
- Run controls
- Visualisation

QLCE units

Number of QLCE units: 1

Unit 7

- Unit 1
- Unit 2
- Unit 3
- Unit 4
- Unit 5
- Unit 6
- Unit 7
- Unit 8
- Unit 9
- Unit 10
- Unit 11

IP address: 192.168.0.57

Version: 0

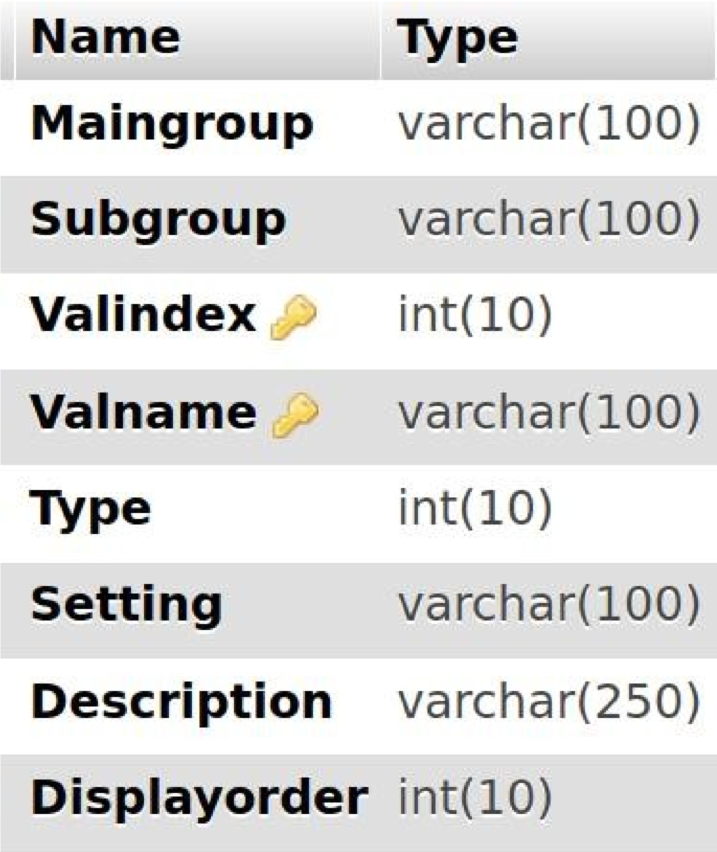
Base AX: 25

Name	Type
Maingroup	varchar(100)
Subgroup	varchar(100)
Valindex 🔑	int(10)
Valname 🔑	varchar(100)
Type	int(10)
Setting	varchar(100)
Description	varchar(250)
Displayorder	int(10)

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

Specify Forth VALUE types in the DB



Name	Type
Maingroup	varchar(100)
Subgroup	varchar(100)
Valindex 🔑	int(10)
Valname 🔑	varchar(100)
Type	int(10)
Setting	varchar(100)
Description	varchar(250)
Displayorder	int(10)

Valname
Name of Forth value type word

Valindex
0= it’s a single Forth VALUE
0<> = it’s indexed value
(VINDEX, STRINDEX etc.)

Type
bool, int, string etc.

A radical alternative to the Windows registry

NOW FOR THE RADICAL BIT

DURING COMPILATION, AT AN EARLY STAGE…

1. Read in the database config file
Now you know how to get to the DB

2. Read in the DB settings table

3. For every **Valname…**

a) Does it exist as a Forth word?
b) If not, dynamically create a VALUE, VINDEX or STRINDEX,
according to **Valindex** and **Type**
c) Set the value from **Setting**

4. You are then free to use these new words in the rest of the compilation process

Euroforth 2020 “Rome”

A radical alternative to the Windows registry

SUMMARY

New Forth words are created, not in code, but from entries in a database table.