RDB for Controls Configuration Management: Complications

Roland Müller

Based on/ Update of previous Presentations given by

B. Franksen: EPICS Meeting `01, SLS/PSI
T. Birke: ICALEPCS `01, San Jose
S. Hunt: IADBG (Int. Accel. DB Group) `01, San Jose

Synopsis, Update + Introduction to next Talk

New Problems Encountered

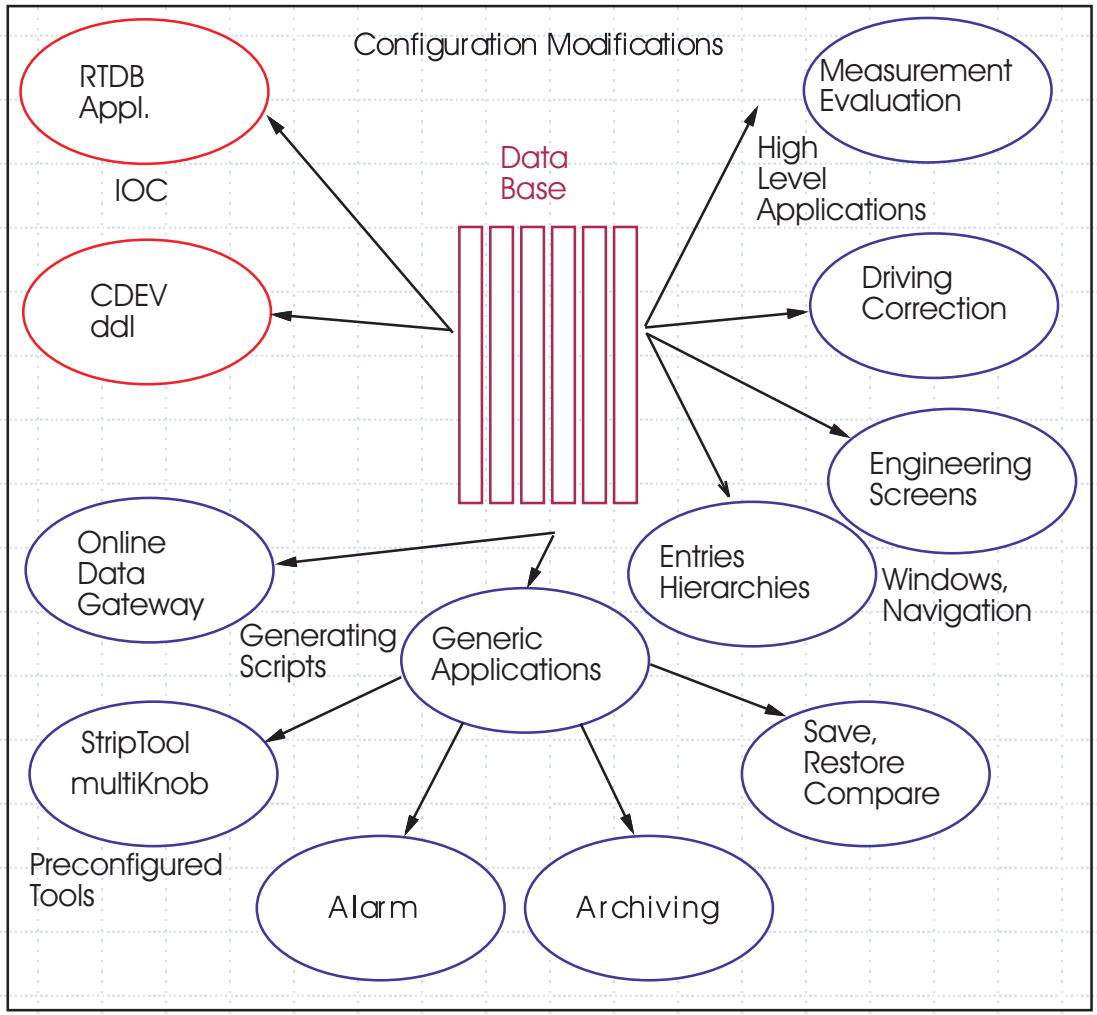
Relational Database for Controls Configuration Management

Thomas Birke

based on work by
T. Birke, B. Franksen, R. Lange, P. Laux, R. M üller
BESSY • Germany

IADBG Workshop • November 2001 • San Jose

Goal: DB Based Configuration Maintenance



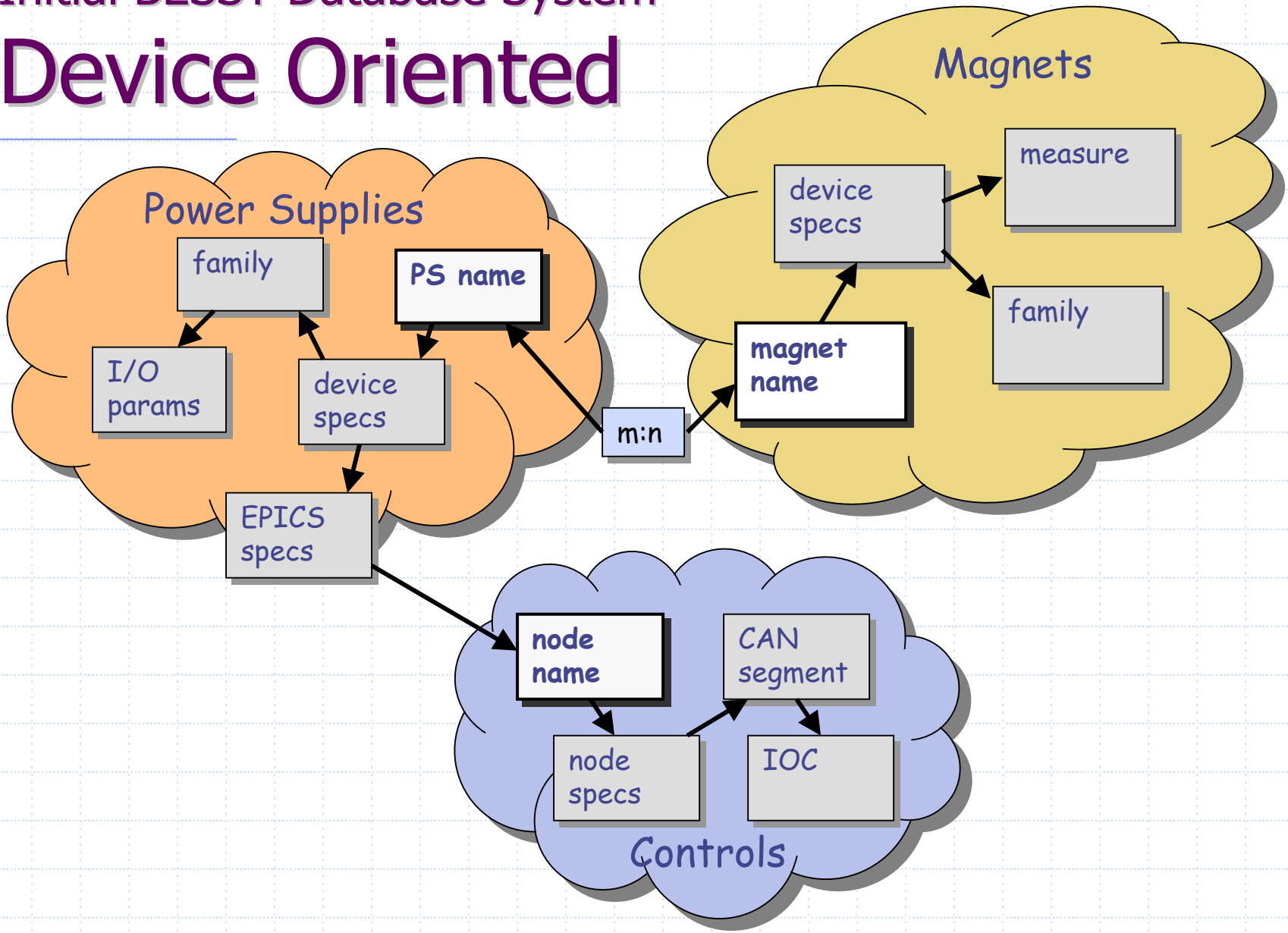
Add/Delete Devices in DB

Procedures and Scripts propagate Changes to Configuration Files

Consistency Guaranteed

Initial BESSY Database System

Device Oriented



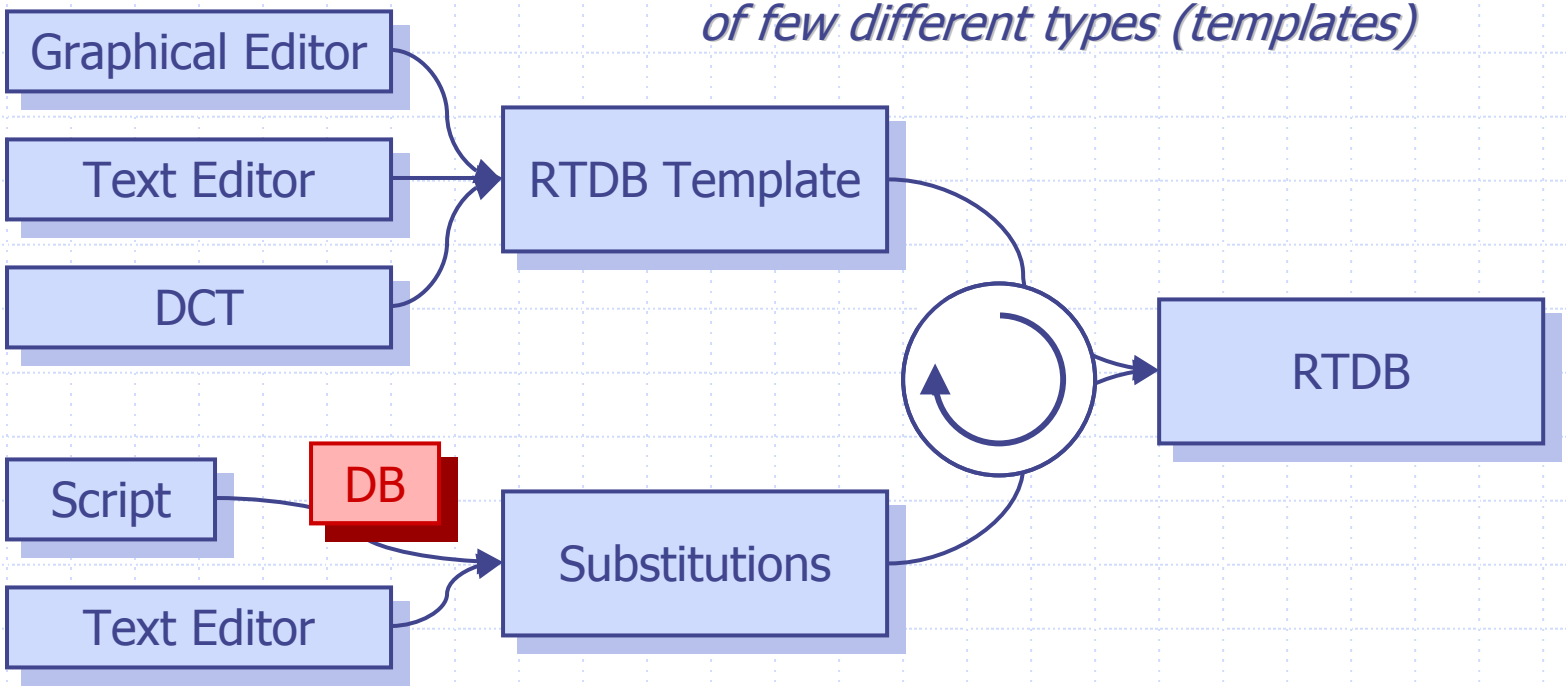
Worked Sufficiently Well for

Creation of RTDB

Standard mechanism

power supplies, vacuum system, RF PLC-interface, triggers/timings, ...

Many instances (substitutions)
of few different types (templates)



Initial / Current State

Database System

- Configuration/parameters are stored and retrieved by device or device class
  (power supply, magnet, rf, vacuum pump…)
- Every class has it's own set of tables/views…
- Devicename is bootstrap
- **<DEVICE>:<channel>.<ATTR>**
  but **<channel>** and **<ATTR>** are not in DB!
  Experience:
- Device oriented model is Unmaintainable:

  Additional knowledge is needed in every DB client to deduce the full channel name

New DB Structure Developed

Basic Elements

- Name

  The central repository of device-names (naming convention)
- Gadget
  - A specific device or a group (class, family) of devices/groups
  - Tree-like structure (leafs are actual devices)

    Higher level nodes provide grouping/abstraction
  - Every gadget has a unique *name*
- Signal
  - The building block of which devices are composed
  - Tree-like structure (leafs are actual signals/records)
  - Roughly corresponds to RTDB template
  - Usually contains one – or few tightly interacting – record(s)
  - e.g. “an analog readback from a CAN-bus I/O-card”

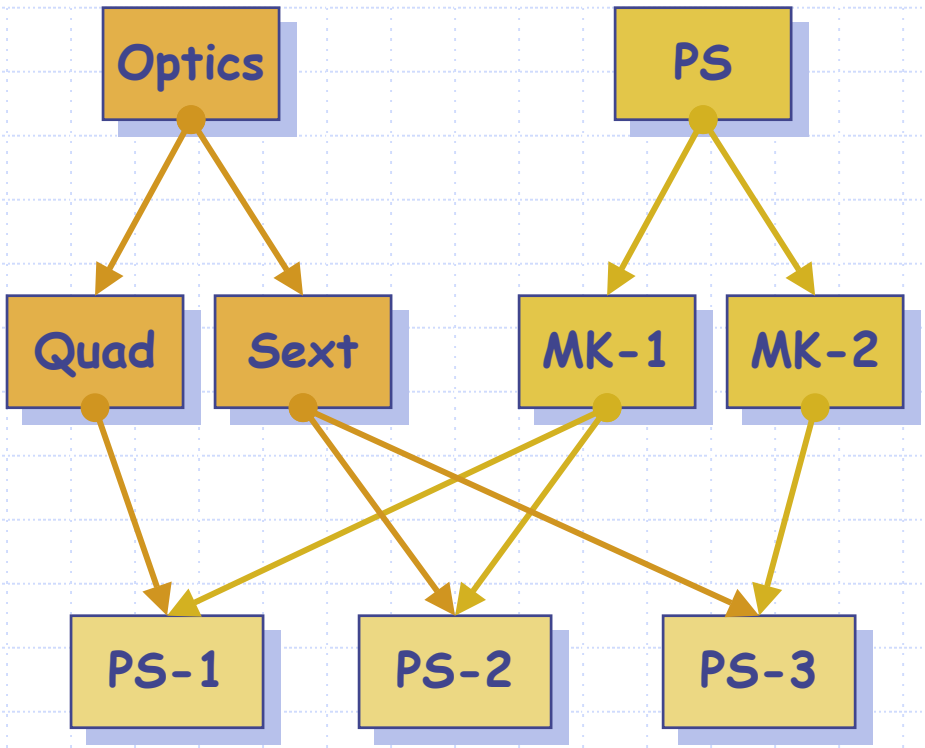
New DB Structure Developed

Basic Elements

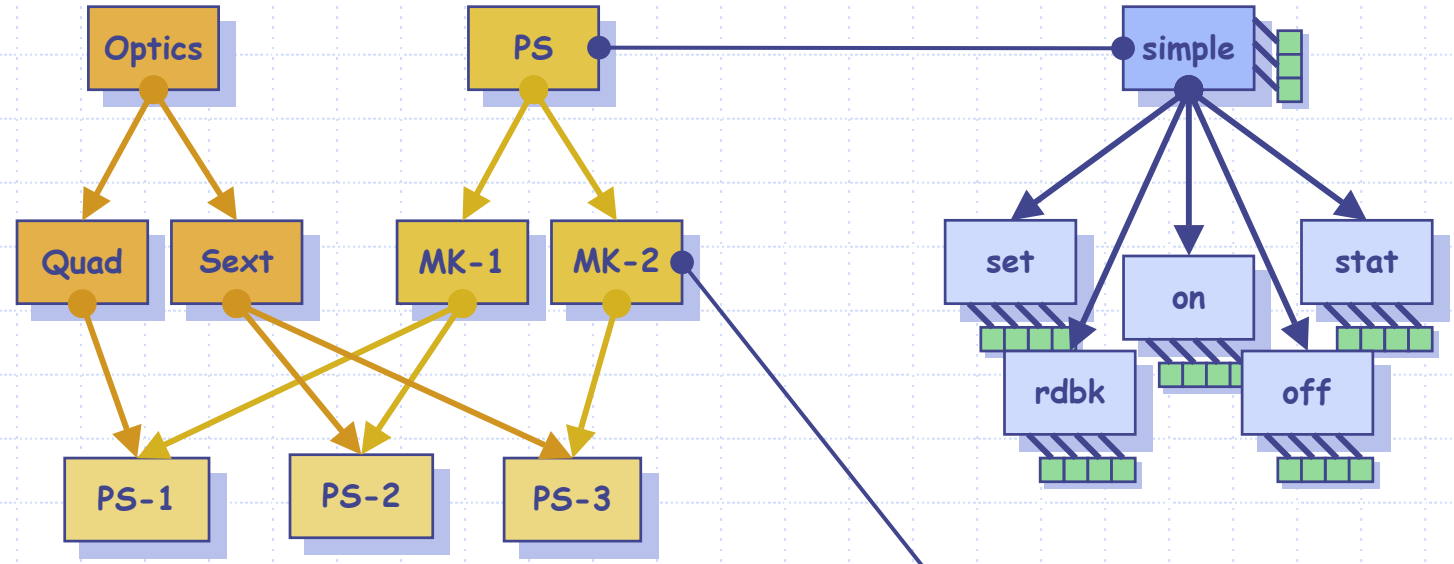
- Attribute
  - Each signal can have a number of attributes
  - Each attribute is related to exactly one signal
  - An attribute of one signal can be "translated" into one or more attributes of a child signal
- Record
  - Instantiates an entity, that can be filled with values
  - Connects gadgets with signals
  - Completely automated table (insert/delete)
- Process Variable (PV)
  - The actual configuration values
  - Connects an attribute with a record and may assign a value

Example (simplified)

Gadgets



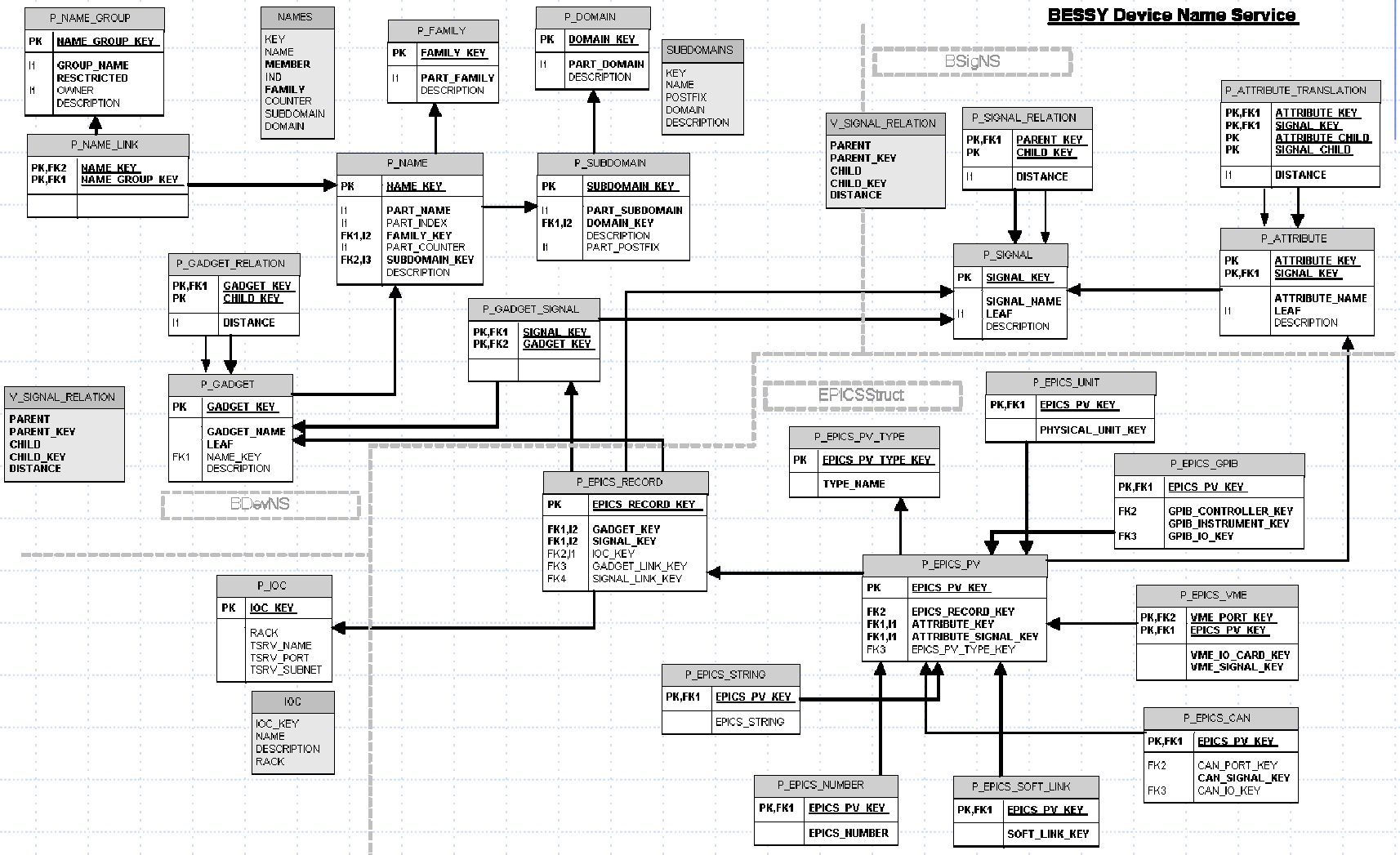
Example (simplified)



<GADGET>:<signal>.<ATTR>

```
PS:rdbk.MUX          ← 2
PS-1:rdbk.MUX        ← 3
PS:simple.EGU        ← "A"
MK-2:simple.MAX-I    ← 350
PS-3:Uout.PREC       ← 5
Quad:alpha.VAL       ← 4.7538
```

New Structural Overview



Project Status as of Nov `01

- Tables and general structure are implemented
  No real data (except names) in those tables yet
- Views, frontends and generic scripts still missing
- Next steps
  - Test system with a new application
  - Develop necessary frontends and scripts in parallel

Topic of Next Talk

- The plan is, to develop generic web browser frontends and standalone command line scripts

New Problems need Solution

- Even our Guineapig `Power Supply´is hard to model
- High Multiplicity / Relative Simplicity should be easy
- Well developed Configuration Management:
  - DB Templates
  - Generic dm2k Screens
  - CDEV ddl Files, Alarms, Save/Restore ...
- Unsolved Problems Remain:
- Problem 1: Complex Devices are set up of Standard Powersupplies (Inheritance)
- Problem 2: Multiple Entries to a Single Physical Device added (Main + Higher Order Input)

Partially Known Problems: Polymorphic Devices

Genome Analysis of Names

HS4	M P	3	D6	R

Magnet/Power Supply Numb. 3 in D6

Horizontal corrector (windings) in Sext 4 of...

(P)KI	K	3	D1	R

Injection Kicker

if K *then*^P ⇒ power supply

UE56	I	D6	R

Insertion Device presently installed in D6

ID UE56 as a complex device with lots of *internal* hardware units

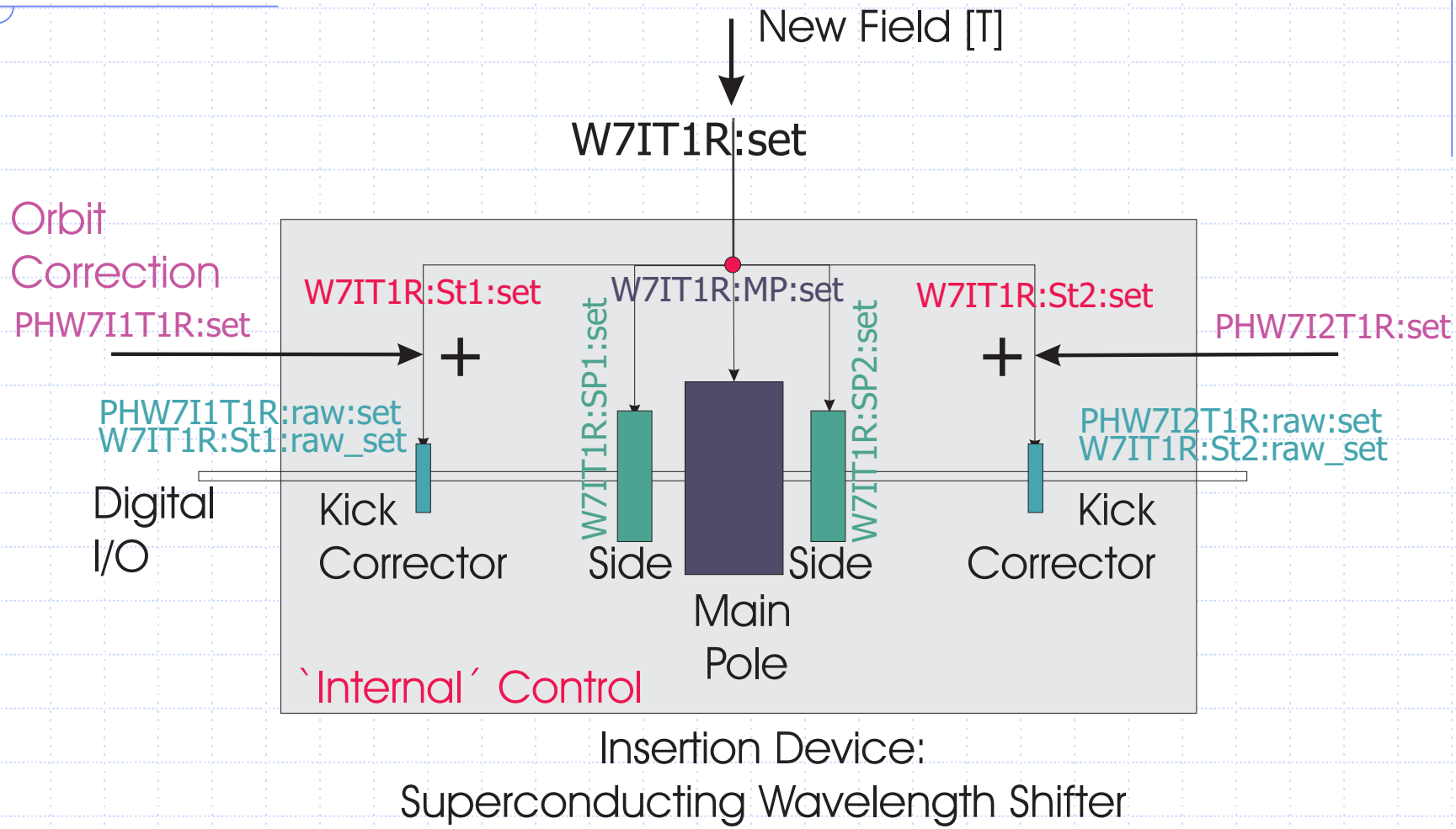
(P)HB UE56	I	2	D6	R

Insertion Device Sub-Unit

if I *then*^P ⇒ power supply of

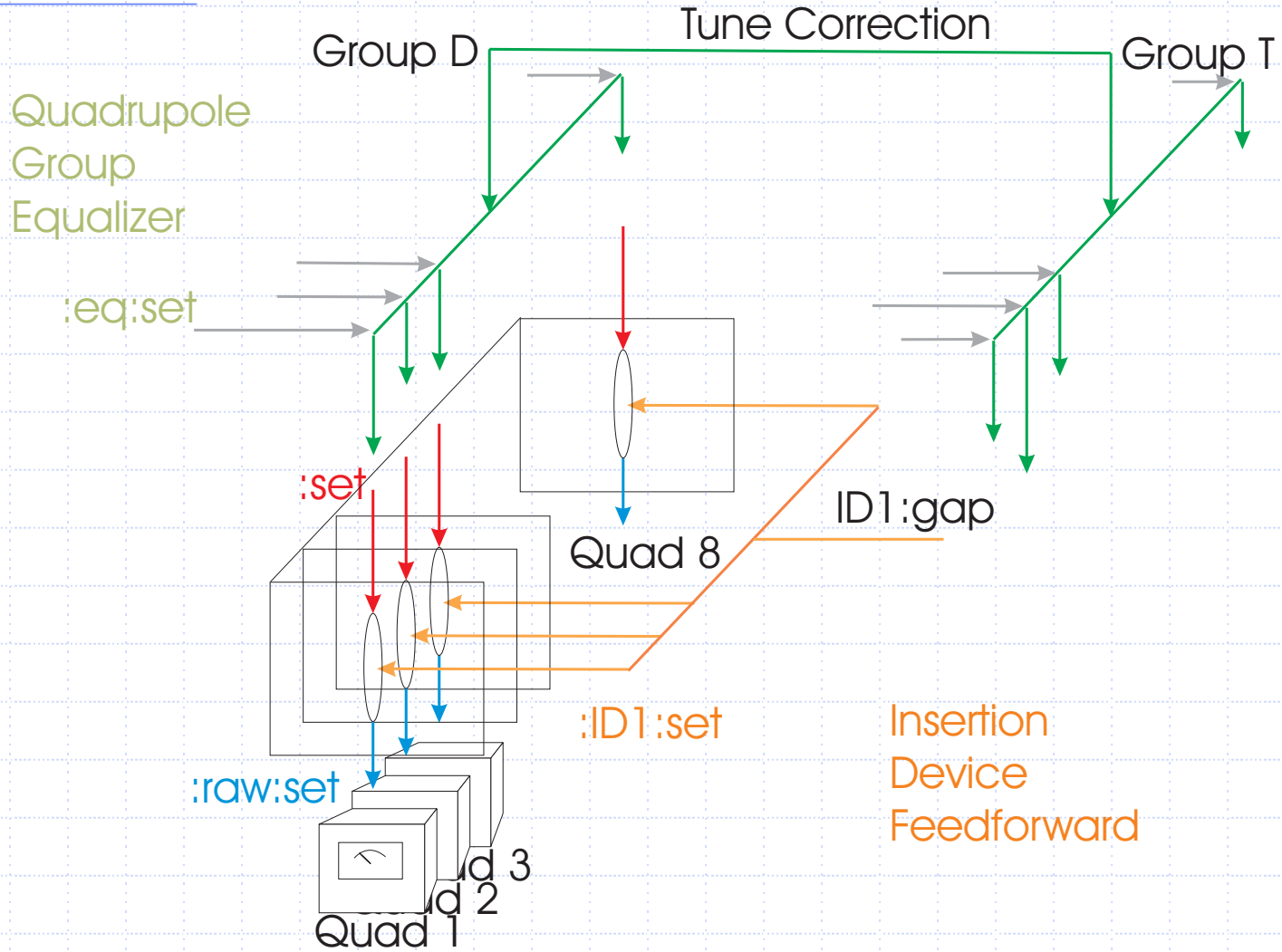
HB ⇒ horizontal UE56bending magnet *external access!*

Composite Device



Insertion Device:
Superconducting Wavelength Shifter

Higher Order Inputs



Summary

- On One Side DB Model, DB Content, Views and Tools should replace the existing Device Oriented System
- On the other End a clear Vision of Generalized Configuration Requirements for all Generic Applications is no more available
- Manpower and Time would help a lot