Cluster 3 – Networked software systems

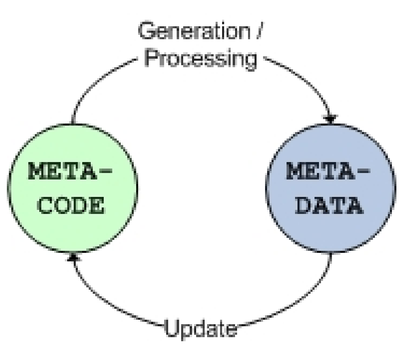
Using Meta-Code for Building Task-Specific WSNs

Igor Talzi, Christian Tschudin / University of Basel, Computer Science Dept
Andreas Hasler, Stephan Gruber / University of Zürich, Dept of Geography
Jan Beutel, Mustafa Yücel, Roman Lim / Computer Engineering and Networks Lab, ETH Zurich

Introduction

Meta-code definition:

- Meta-code deals with the meta-data, or
- Meta-code describes behaviour, the actual implementation is still based on the native code.



The potential use:

- Auto-configuration (react to changes in topology, sensor attachments, etc)
- Self-documenting (report changes)
- Proactive actions

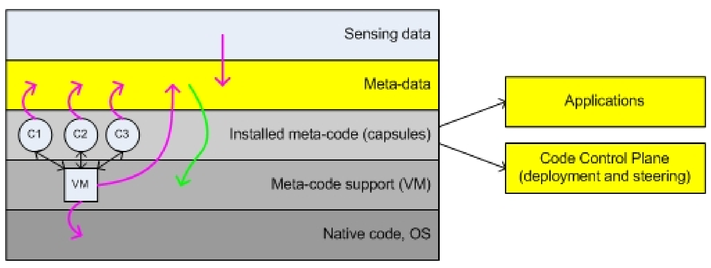
Code/Data «Sandwich»

Mainly, we focus on

- Meta-data management,
- Code Control Plane [CCP] for code deployment and steering,
- Meta-code compression.

Meta-data management:

- Network configuration (topology, faults, etc)
- Software versioning
- Instruction set dictionary
- User-defined structures



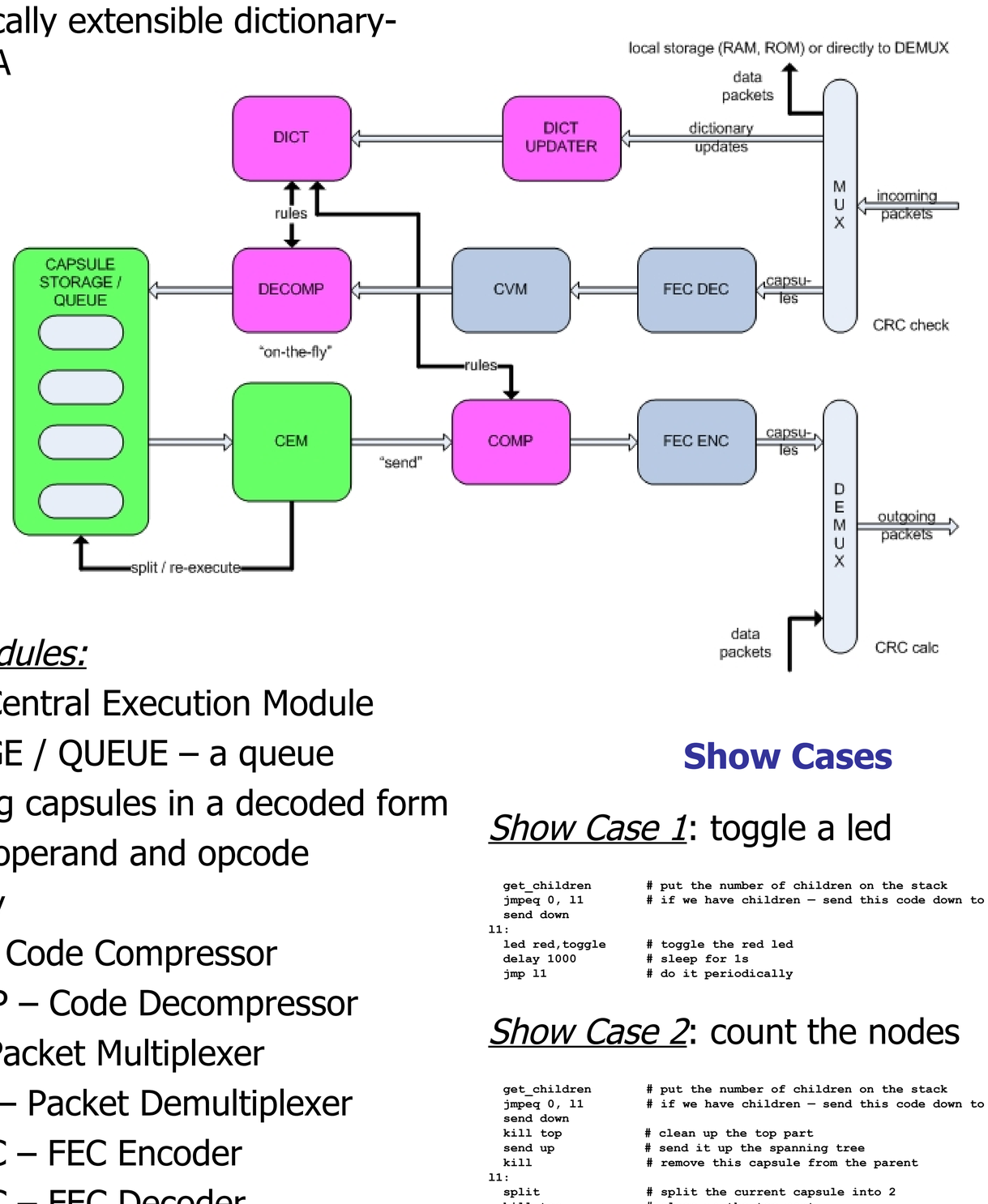
Code Control Plane [CCP]:

- Track changes, on-demand requests
- Uses meta-data layer for storage
- Creates reports

Node Architecture

Basic principles:

- Stack-based VM
- Shared memory for vars (heap)
- Memory protected capsule execution
- FIFO-like scheduling, blocking operations (e.g. «delay N» or «send_dt up»), direct access to the queue
- Errors are reported to the meta-data layer
- Dynamically extensible dictionary-based ISA



Basic modules:

- CEM – Central Execution Module
- STORAGE / QUEUE – a queue containing capsules in a decoded form
- DICT – operand and opcode dictionary
- COMP – Code Compressor
- DECOMP – Code Decompressor
- MUX – Packet Multiplexer
- DEMUX – Packet Demultiplexer
- FEC ENC – FEC Encoder
- FEC DEC – FEC Decoder
- CVM – Code Validation Module

Instruction Set

Basic instructions:

- Stack ops («push», «pop»)
- Heap ops («read», «write», «move», etc)
- Arithmetic ops («add», «mult», etc)
- Binary\Logic ops («and», «xor», etc)
- Jumps («jmpeq 7,L1», «jmp L2», etc)
- Debug («led green,off»)
- Macro-instructions

Macro-Instructions

Command	Semantics
send_dt/code up/down	Send the current capsule (CODE) or the stack content (DATA) up/down the spanning tree
get_meta_dt F	Get meta-data field F
mod_meta_dt F,V	Modify/create meta-data field F with value V
split	Split the capsule into 2 at the current position
merge A,B	Merge capsules A and B
copy L,ADDR	Copy the code from the current position to the label L and send it to the address ADDR (broadcast, unicast)
kill "/top/bottom	Remove the entire capsule, or its top, or a bottom part
wr/rd/rm_dict	Write/read/remove a record from the instruction dictionary
ping ADDR	Ping a node
get_children	Get the list of available children nodes

Show Cases

Show Case 1: toggle a led

```
  get_children        # put the number of children on the stack
  jmpeq 0, 11         # if we have children – send this code down to them
  send down
11:
  led red,toggle      # toggle the red led
  delay 1000          # sleep for 1s
  jmp 11              # do it periodically
```

Show Case 2: count the nodes

```
  get_children        # put the number of children on the stack
  jmpeq 0, 11         # if we have children – send this code down to them
  send down
  kill top            # clean up the top part
  send up             # send it up the spanning tree
  kill                # remove this capsule from the parent
11:
  split               # split the current capsule into 2
  kill top            # clean up the top part
  send up             # send it up the spanning tree
```

(the code for counting incoming capsules at the top of the tree is not shown)

Code Compression

`ping ADDR`	`push 3`
	`11:`
	`get_nodeid`
	`report ADDR`
	`delay 1000`
	`dec`
	`jmpneq 0,11`

`report ADDR`	`get_meta_dt ADDR` `send_dt up`

Future Work

- Obtaining initial results on simulation
- More complex show cases (e.g. skew-balance time sync protocol)
- Power profiling

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

FNSNF
FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

Bundesamt für Umwelt BAFU
Office fédéral de l'environnement OFEV
Ufficio federale dell'ambiente UFAM
Uffizi federal d'ambient UFAM

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

UNI BASEL

iat/MICS-Lausanne'09