FlexCup: A Flexible and Efficient Code Update Mechanism for Sensor Networks

Pedro Jose Marron, Matthias Gauger, Andreas Lachenmann et al.

EWSN'06

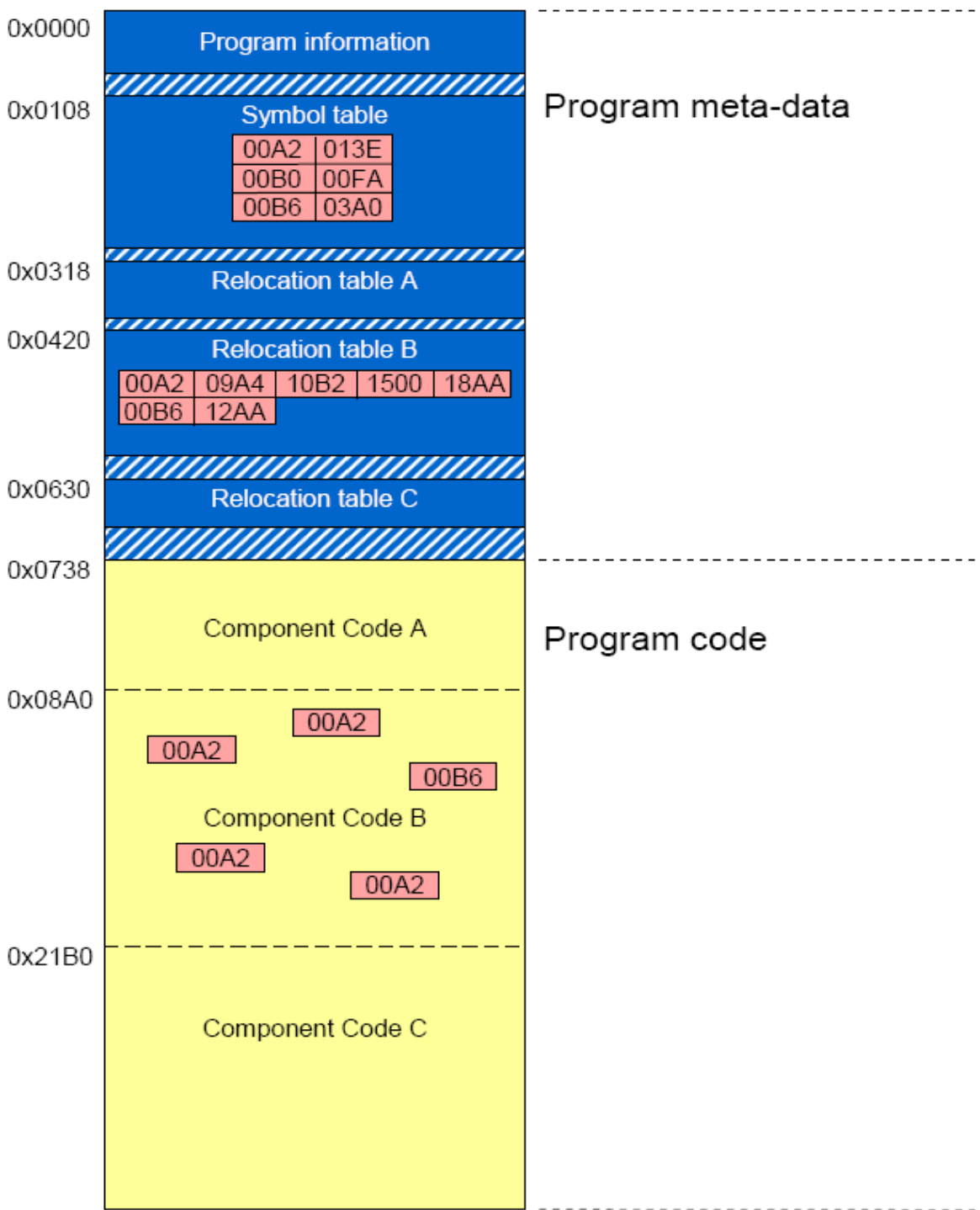
Presented by lin,yu-chieh

Outline

- Introduction
- FlexCup
  - Component and Meta-Data Generation
  - Runtime Linking
- Experimental Evaluation

Introduction

- FlexCup
  - Enable on fly reinstallation of software components in TinyOS-based sensor nodes.
  - Is able to reconfigure, exchange or reinstall parts of an application.
  - Two phases
    - Code generation phase.
    - Linking phase


0x0000
Program information
0x0108
Symbol table
00A2 013E
00B0 00FA
00B6 03A0
Program meta-data
0x0318
Relocation table A
0x0420
Relocation table B
00A2 09A4 10B2 1500 18AA
00B6 12AA
0x0630
Relocation table C
0x0738
Component Code A
Program code
0x08A0
00A2
00A2
00B6
Component Code B
00A2
00A2
0x21B0
Component Code C

FlexCup

- FlexCup generates meta-data that describes the compiled compnents.
- Meta-date
  - Place the new component inside the application.
  - Relink function calls.
  - Address binding of data.

FlexCup

- **Component Generation**
  - TinyOS applications consist of a set of system and application component that are ‘wired’ to generate a running program.
  - It is hard to replace only a part of the compiled program.

FlexCup

- nesC 1.2 allows compiling a set of nesC components into a separate object file , called *binary component*.
- Ex.
  - Radio communication.
  - applications components.

FlexCup

- **Meta-Data Generation**
  - Three parts
    - Generic program information.
    - Program-wide symbol table.
    - Relocation table.

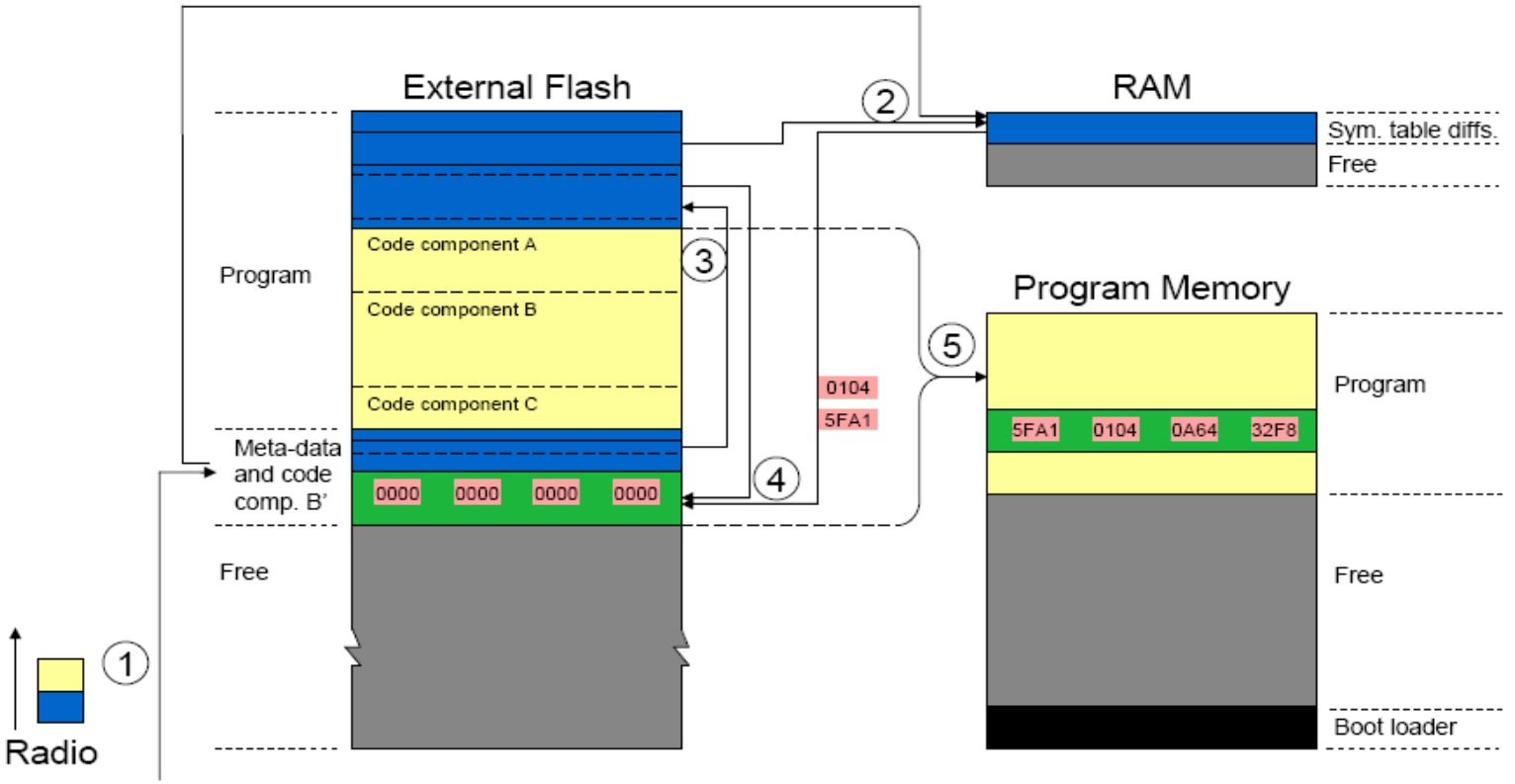
FlexCup

- **Optimization**
  - Symbols in the symbol and relocation tables are identified by a two-byte id.
  - Compress the size of the relocation tables by combining entries with the same id .

FlexCup

- Runtime Linking
  - Storage of code and meta-data
  - symbol table merge
  - relocation table replacement
  - reference patching
  - Installation and reboot

FlexCup


External Flash
RAM
Program
Code component A
Code component B
Code component C
Meta-data and code comp. B'
Free
0000
0000
0000
0000
Sym. table diffs.
Free
Program Memory
Program
5FA1
0104
0A64
32F8
Free
Boot loader
0104
5FA1
Radio
1
2
3
4
5

FlexCup

- Runtime Linking
  - Storage of Code and Meta-Data
    - FlexCup Basic
      - Transfers the whole binary components and its meta-data.
    - FlexCup Diff
      - Only transfers the incremental changes between the new binary component and the one already stored on the node.

FlexCup

- Symbol Table Merge
  - Combine the existing program symbol table with the newly received symbol data.
  - Needs to calculate the storage of the variables in data memory and needs to set the symbol addresses accordingly.

FlexCup

- Relocation Table Replacement
- Reference Patching
  - Going through the entries of the relocation tables of all components.
  - Checking whether any of the references needs to be updated.
- Installation and Reboot.

Experimental Evaluation

Table 1. Complexity of sample applications

Applications	Size (bytes)	Number of nesC components	Number of binary components
`OscilloscopeRF`	11784	39	6
`Surge`	17096	53	10
`AcousticLocalization`	24272	69	15

Table 2. Changes performed on the applications

Application	Class	Code Update
`OscilloscopeRF`	small	global constant
`OscilloscopeRF`	small	additional call
`OscilloscopeRF`	small	sensor reading
`Surge`	internal	function exchange
`Surge`	internal	wiring configuration
`AcousticLocalization`	external	component exchange

Experimental Evaluation

Table 3. Average size of code update algorithms

	Program Code Size (bytes)		
Application	Deluge	MOAP-Diff	FlexCup
OscilloscopeRF	10868	16742	26715
Surge	11326	17213	27466
AcousticLocalization	10650	16728	26692

Table 4. Size of components and meta-data (in bytes)

	Transmitted Data Size								Flash Memory Data Size			
		MOAP-	FlexCup Basic			FlexCup Diff				MOAP-	FlexCup	
Code Update	Deluge	Diff	Meta	Code	Total	Meta	Code	Total	Deluge	Diff	Basic	Diff
global const.	23142	11	799	1198	1997	530	15	545	23142	28538	37337	35885
additional call	23142	1230	801	1202	2003	760	5	765	23142	28542	37343	36105
sensor reading	23142	2835	537	886	1423	523	114	637	23142	28608	36743	35977
function exch.	28652	7684	1056	3258	4314	1110	1587	2697	28652	33440	43561	41944
wiring config.	28652	375	1355	2142	3497	1290	8	1298	28652	34272	42744	40545
comp. exch.	34162	7802	2565	4773	7338	2611	532	3143	34162	40156	58014	53736

Experimental Evaluation