

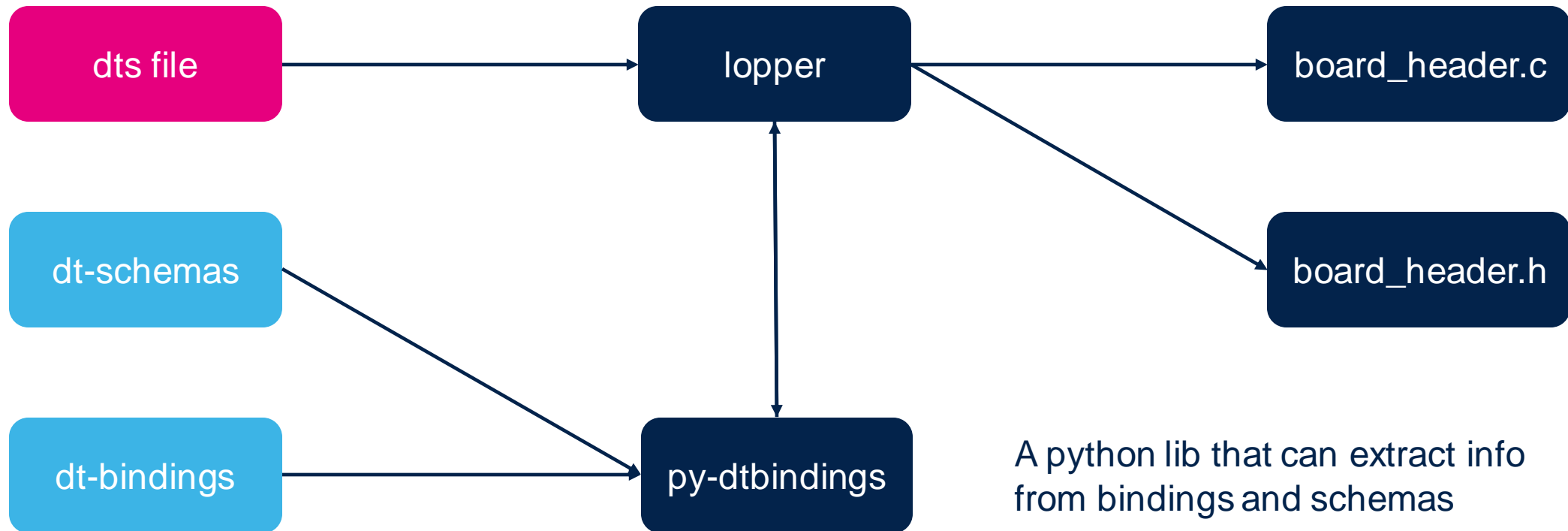


life.augmented

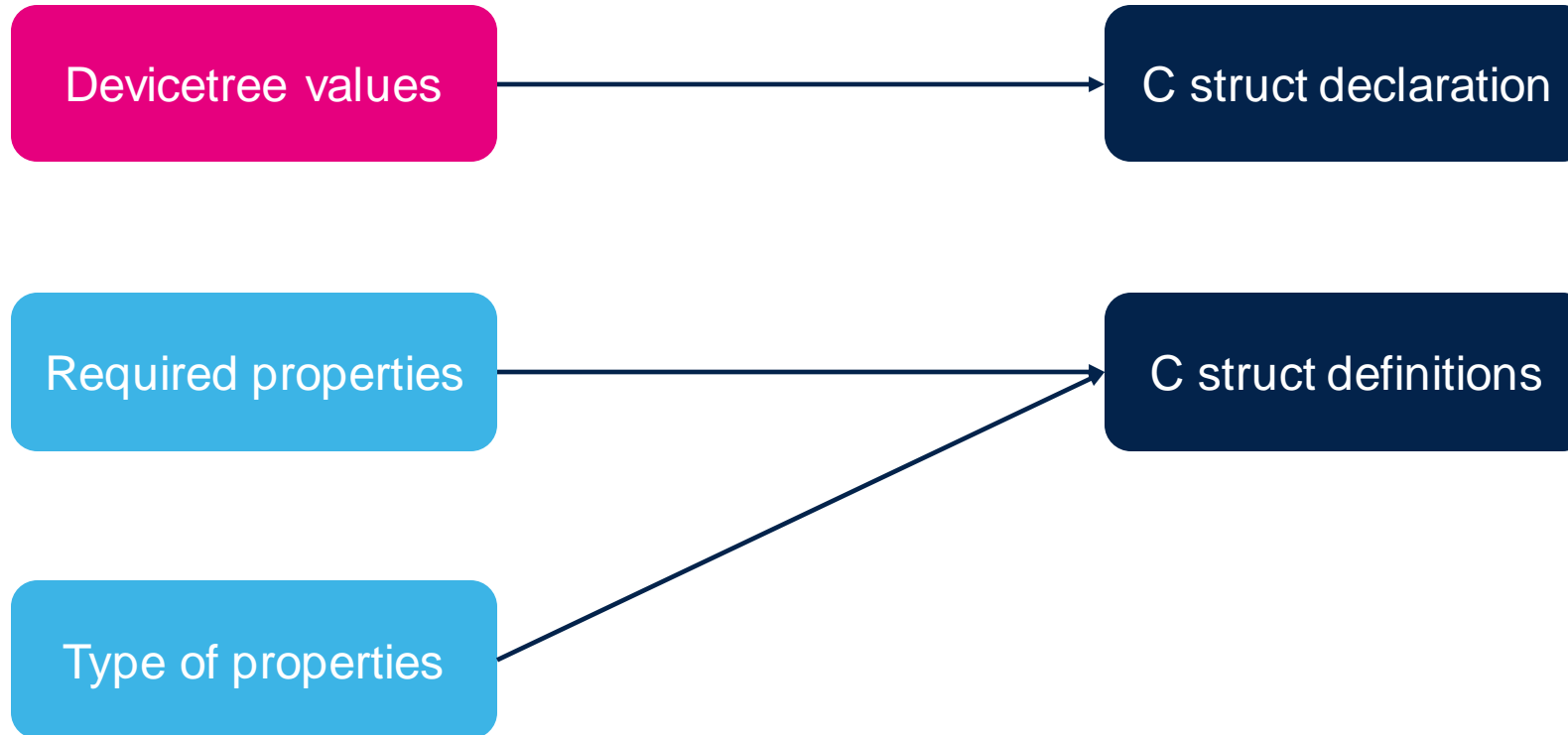
Baremetal configuration via devicetree using lopper

Valentin MONNOT

Workflow



Workflow



Example

dt-bindings/st,stm32-uart.yaml

```
109   required:
110     - compatible
111     - reg
112     - interrupts
113     - clocks
```

dt-schemas/clocks.yaml

```
clocks:
  anyOf:
    - type: object # for nodes named 'clocks'
    - $ref: "/schemas/types.yaml#/definitions/phandle-array"
```

dt-schemas/dt-core.yaml

```
reg:
  $ref: "types.yaml#/definitions/uint32-matrix"
```

dt-schemas/interrupts.yaml

```
interrupts:
  $ref: "types.yaml#/definitions/uint32-matrix"
```

Generated struct

```
struct serial_s{
    const void * rcc;
    const uint32_t * interrupts;
    const uint32_t * reg;
};
```

Example

```
uart4: serial@40010000 {  
  
    clocks = <0x06 0x97>;  
    compatible = "st,stm32h7-uart";  
    interrupts-extended = <0x12 0x1e 0x04>;  
    phandle = <0x90>;  
    pinctrl-0 = <0x1c>;  
    pinctrl-1 = <0x1d>;  
    pinctrl-2 = <0x1e>;  
    pinctrl-names = "default", "sleep", "idle";  
    reg = <0x40010000 0x400>;  
    status = "okay";  
    wakeup-source;  
};
```

```
const uint32_t INTERRUPTS_SERIAL_40010000[3] = {0x12 , 0x1e , 0x04};  
const uint32_t REG_SERIAL_40010000[2] = {0x40010000 , 0x400};  
const SERIAL_S SERIAL_40010000 = {  
    .rcc = NULL,  
    .interrupts = INTERRUPTS_SERIAL_40010000,  
    .reg = REG_SERIAL_40010000,  
};
```

- [WIP] Generate struct for phandle nodes
- How to process optional nodes ?
- Any comments or suggestions?