Energy Framework

Laura Marie Feeney[1] Daniel Willkomm[2]

[1]Communication Networks and Systems Lab
Swedish Institute of Computer Science

[2]Telecommunication Networks Group
Technische Universität Berlin, Germany

Overview

- extensible OMNeT++ framework for modeling energy consumption
- battery as a first-class component of simulation
- mostly “pure” OMNeT++, some functionality is framework specific (mobility-fw and MiXiM)
- outline
  - battery modeling
  - design issues

Goals

- include battery as active element in simulation
  - host failure due to battery depletion
  - energy-aware protocols & state-of-charge estimation
  - voltage-dependent operation
  - energy harvesting
- support for models with varying level of detail
- support for development of sophisticated battery models for discrete event simulation (hard!)

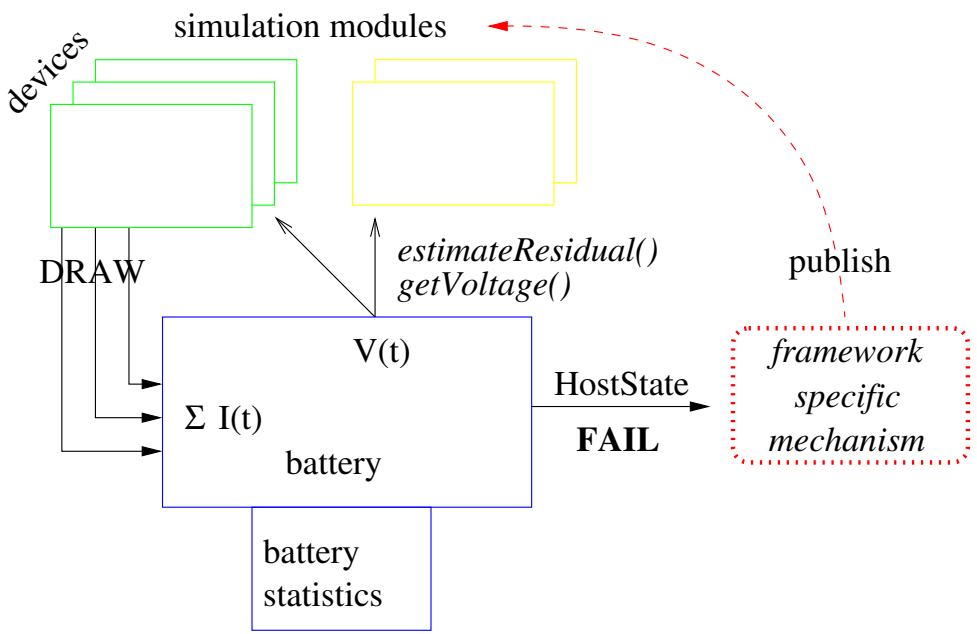
Battery model

- reality: complex electro-chemical reactions
- empirical, stochastic, circuit, analytic models
  - hard to incorporate directly into DES
  - need real time model, *very* fast (one per node)
- linear: capacity = mA×hr; draw = mA × *time*
  - operation counting
- target model: output voltage
  - discharged when V_{out} below threshold
  - include key non-linear effects

Energy Framework

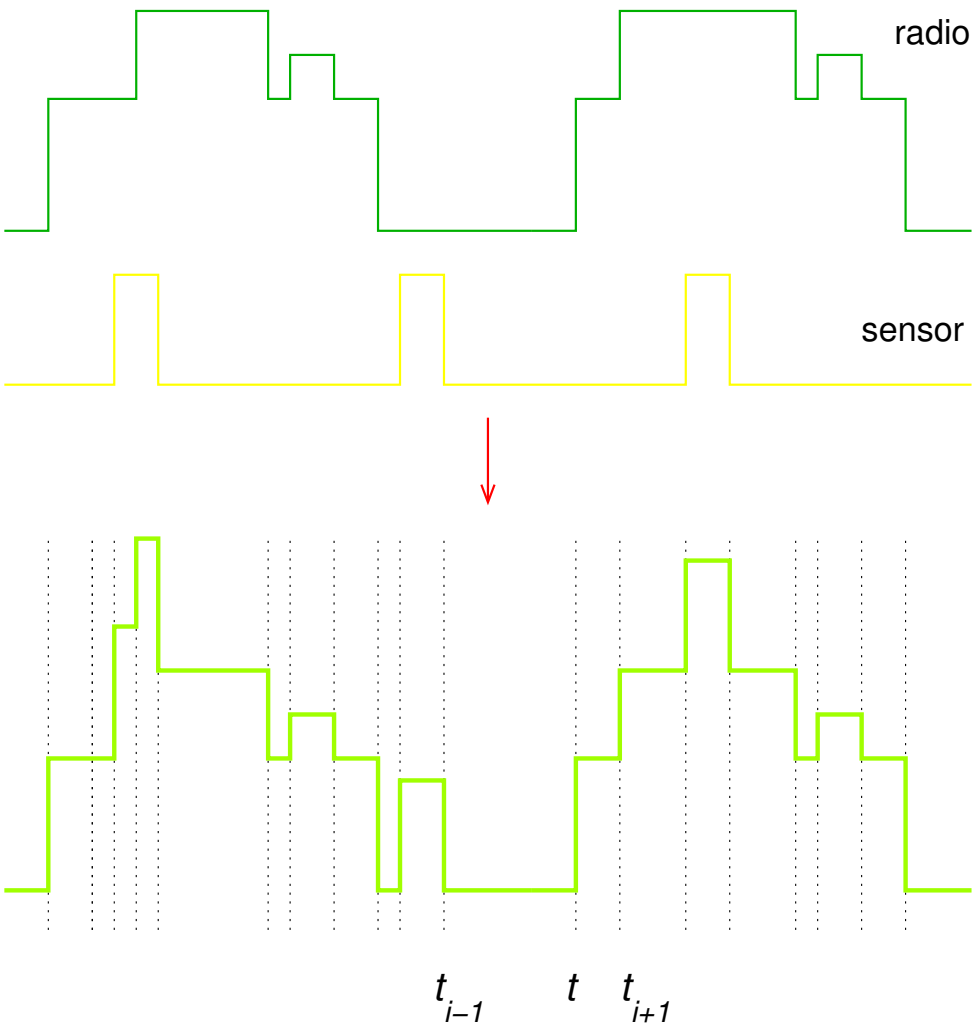
- separate modeling of battery and energy consuming operations
  - modular, extensible
  - multiple sources of energy consumption/generation
  - variety of abstractions for energy consumption/generation
  - explicit state-of-charge estimation
  - flexible handling of statistics
- clean handling of HostState notification
- facilitates experimentation with new battery models

Design I



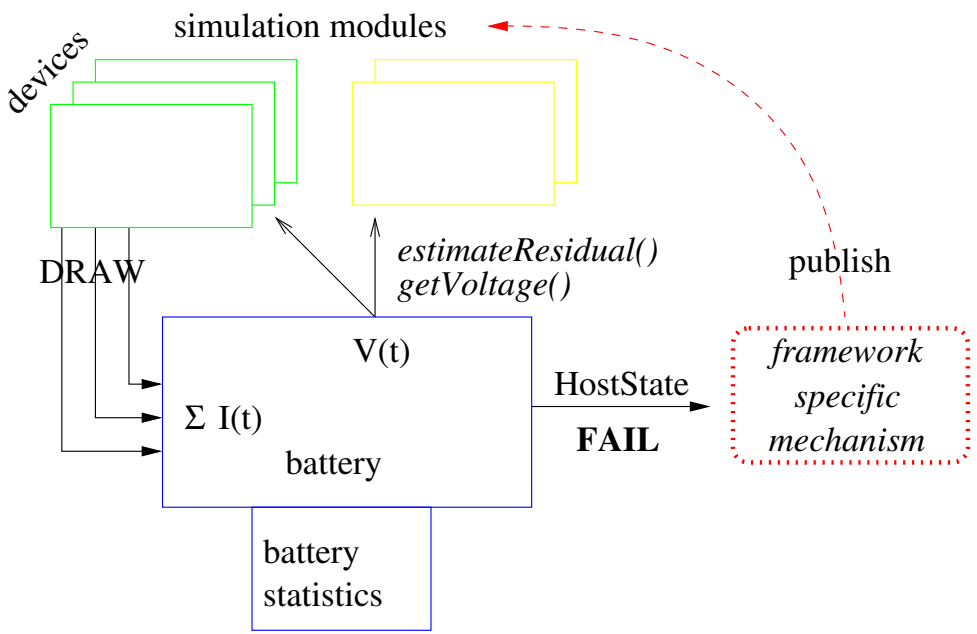
- linear: $V_{nominal} \times \sum_d I_d \times (t_i - t_{i-1})$
- also discrete decrements
- statistics are handled in separate module
- validated against BatteryModule2.0

Design I



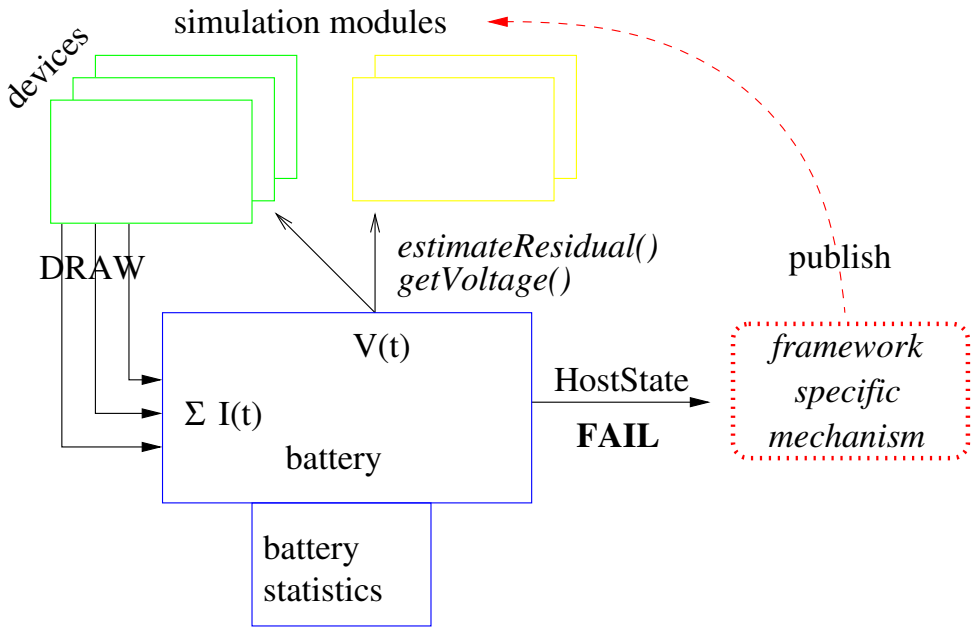
- linear: $V_{nominal} \times \sum_d I_d \times (t_i - t_{i-1})$
- also discrete decrements
- statistics are handled in separate module
- validated against BatteryModule2.0

Design I



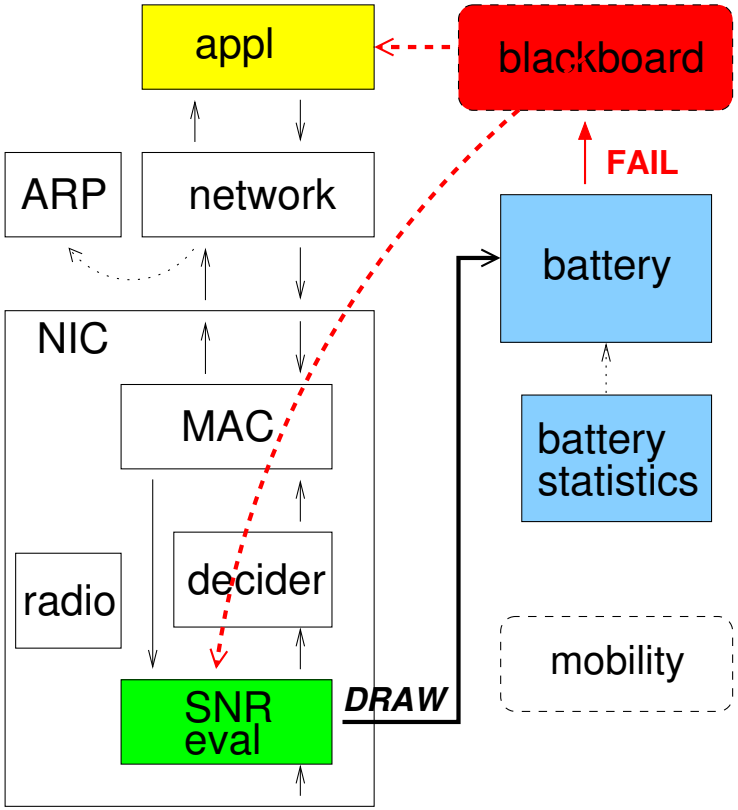
- linear: $V_{nominal} \times \sum_d I_d \times (t_i - t_{i-1})$
- also discrete decrements
- statistics are handled in separate module
- validated against BatteryModule2.0

Design II



- "device" - any module that consumes energy
- DRAW - uniform interface to the battery module
- *estimateResidual()* - get state from the battery module
- *getVoltage()* - get output voltage

Design III



- HostState notification
- non energy-aware modules must handle
- framework specific
  - blackboard (pub/sub)
  - enforced in MiXiM (with opt-out)
  - use of signals (?)

Conclusions

- Energy Framework
  - extensible, modular
- current status
  - multiple devices
  - linear battery model
  - estimateResidual() (follows battery model)
  - HostFailure notification
  - statistics module
  - validation tests
- future
  - support for development of complex battery models