Internet of Things
2017/2018

Life Cycles and Domains

Johan Lukkien
Leila Rahmann

John Carpenter, 1982
Questions

• What is the life cycle of IoT systems and components?
• What is the impact of the application domain on these life cycles?
The life cycle of a product or system is the series of stages it goes through from inception to decline.

A typical life cycle for a software system is given to the right.

More detail is obtained by adding information regarding the activities in the stages.

Notes:
- Also system parts have life cycles, affecting the overall system.
- The life cycle should also address evolution, redesign.
Life cycles for IoT

• IoT life cycles pertain to
  – devices,
  – (software) components & services
  – applications

• IoT applications are networked
  – distributed system (programming)
    • needs concept development and tooling support
  – cumbersome deployment/commissioning
  – inevitably increasing points of failure
    • require machine intelligence both in deployment as well in self management during the life cycle

• Concrete life cycles differ per domain
  – Home
  – Office
  – Outdoor
    • city
    • field – e.g. crops
    • mobile – vehicles, people
    • ad-hoc (emergency response)
  – Industry

• Life cycle analysis is key in understanding architectural requirements
IoT Device life cycle

- Life cycles have a generic structure but are different for each device type

- Examples:
  - **phone**: deployment is through purchasing; other commissioning is through manufacturer. Life cycle server is with connectivity provider or manufacturer.
  - **office IoT**: commissioning is by an installation company. Life cycle server is with owner or with manufacturer.

- Important aspects concern responsibilities and control by involved stakeholders
  - in particular: responsibility for software updates
IoT: involved software types

- Embedded Operating System / runtime executive, middleware
  - typically installed as part of a firmware image
  - supporting the running of components, applications
- Libraries
  - e.g. a CoAP library, linked into an executable
- (Runnable) components exposing services
  - e.g. a CoAP based service for inspecting the temperature or adjusting the heater
- Applications (application components)
  - e.g. a management application using temperature services and controlling the heater
  - e.g. a data analytics application
IoT: software update packaging

- **Firmware**
  - full update of a node’s software
- **Module**
  - a library or application component, possibly even an OS or system part
- **Setting**
  - parameter settings on the existing system
IoT service and component life cycle

- Components deliver services, or take part in that
- Example:
  - **temperature sensor**: a component implementing a temperature service is installed on the sensor node over the air
- Recommissioning is triggered by renewal of libraries, of versions
- Important again are **responsibilities and control** by involved stakeholders
  - in particular: responsibility for software updates after first install
  - also of commissioning, service publication (repository)

**CONSTRUCTION**
- Service component development:
  - Specification
  - Design
  - Implementation
  - Testing

**DEPLOYMENT**
- Install service component on IoT devices
- Configure service
- Register or publish service for discovery

**UNDEPLOYMENT**
- Unregister or unpublish service from discovery
- Remove service component from device

**OPERATION**
- Handle service requests

**DESTRUCTION**
- End of life
IoT Application life cycle

- Life cycles of applications may depend on life cycles of components, libraries
- Examples:
  - thermostat application: searches and finds services that yield temperature and control services that control the heater (needs further details and access control of course)
- Important aspects concern access control by involved stakeholders
Characteristics of the home domain

- The home is in principle unmanaged
  - one-time configuration of new equipment
    - legacy effects: a very old device requiring obsolete security protocols
    - default configuration not altered
    - wide variety of policies and mechanisms for installation
  - responsibility for software updates not assigned
    - devices difficult to reach after installation
    - automatic update by manufacturer is a security and functionality hazard
  - problems and side effects very difficult to understand
    - lack of conceptual model
    - lack of management addressing system level concerns like data protection and service quality
    - no control over data
  - no money available for extra management services
The office domain

• The office is managed
  – central access control, policies
  – clear procedures and responsibilities for system updates
  – extensive standardization is possible
  – higher cost acceptable

• Conflicting concerns of stakeholders
  – ‘BYOD’ interferes
  – Data management
    • office manager has details about users, visitors
Other domains have further characteristics

• City, mobile:
  – services provide access to infrastructure
  – ownership needs attention

• Industry
  – single owner of data
  – very controlled environment
  – production data integrated with process improvement, maintenance and logistics

• These characteristics:
  – alter the implementation of the life cycles
  – lead to very different problems in life cycle stages
Questions

• What is the life cycle of IoT systems and components?
• What is the impact of the application domain on these life cycles?