(Sub)system Level Architecture
&
Requirements

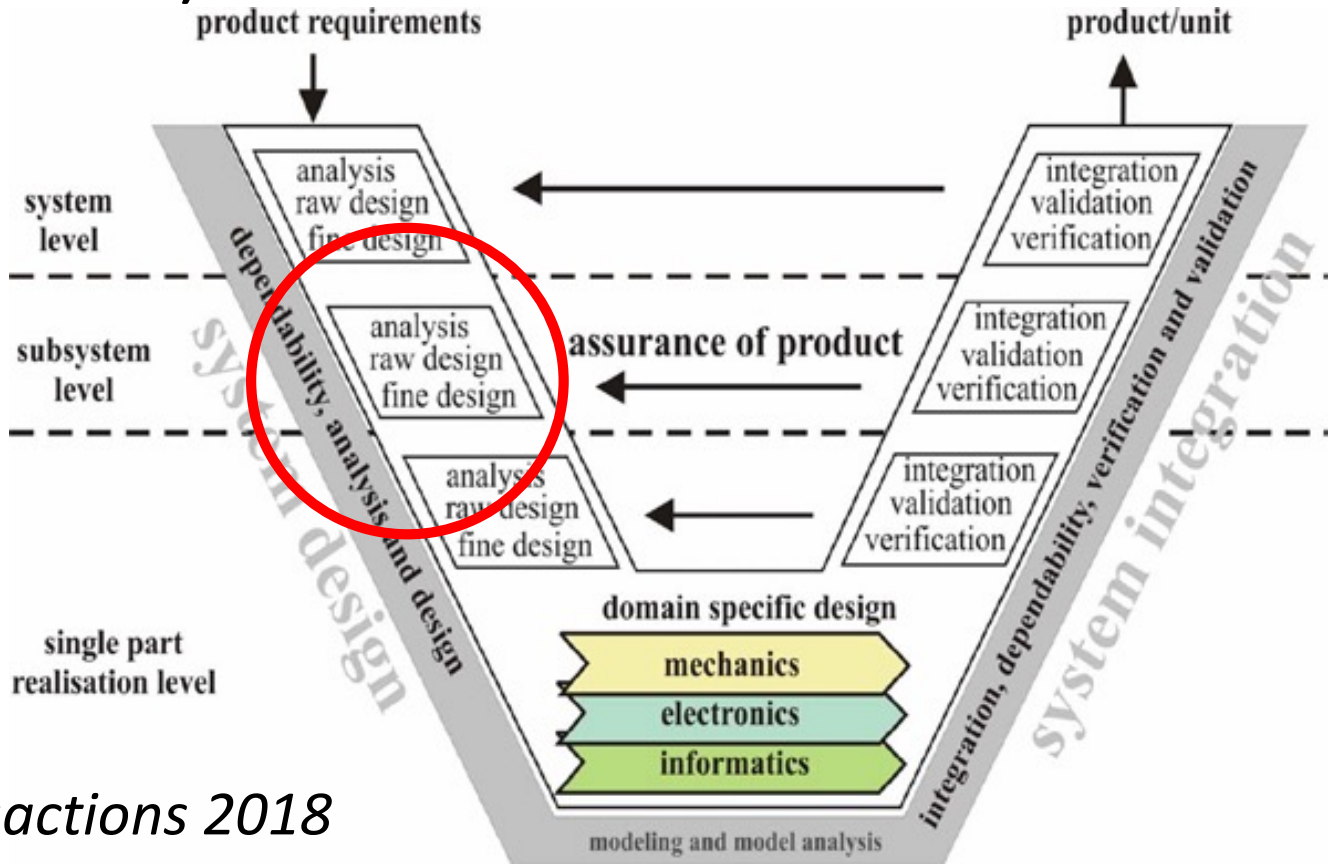
Proyectos Integrados

Grado en Ingeniería Electrónica, Robótica & Mecatrónica

Profesor: José Ángel Acosta Rodríguez (jaar@us.es)

The V-Model

VDI 2206 (2004)



In FME Transactions 2018

Requirements (recall)

Requirement definition (EIA 632):

"Something governing

- @ Functional: WHAT?
- @ Performance: HOW WELL?
- @ Environmental: UNDER WHAT CONDITIONS?

a product will achieve a given purpose"

Requirement types:

- @ Users
- @ System

Sub-sytem Level - Architecture

- An architectural model presents an abstract view of the sub-systems making up a system
- May identify different types of functional component in the model
- May include major information flows between sub-systems
- Block diagram between sub-systems

Holistic approach (recall)

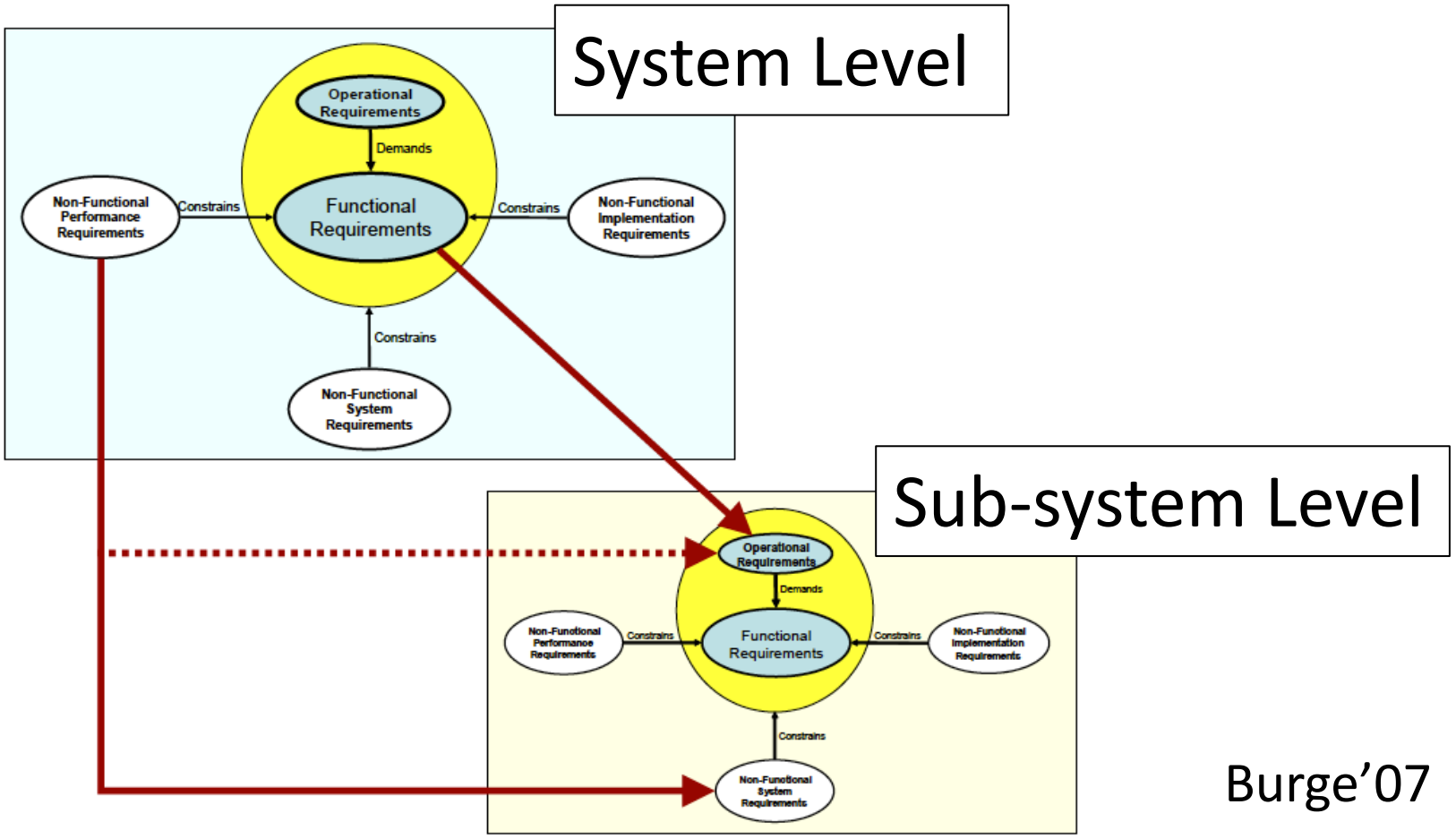
- Operational (O): major purporse (WHAT)
- Functional (F): How to do it to achieve O (HOW)
- Non-Functional (NF): Constraints (UNDER COND.)
  - Performances (NF-P): associated to a F requirement
  - System (NF-S): constraints affecting the whole system:
    - Physical: size, weight, style
    - -bilites: relia-, maintaina-, interopera-, deploya-
    - Performance: speed, manoeuvrability, cost
    - Commercial/contractual: e.g. ready for trials on a date
  - Implementation (NF-I): built with a specific technology

Holistic approach

Civil Aircraft example:

- O: Transfer passengers and baggage from one point to another safely
- Functional:
  - Navigate (F1)
  - Control flight (F2)
  - Store passengers and baggage (F3)
  - Communicate with others and ATC (F4)
  - Propulsion (F5)
- Non-Functional (NF):
  - NF-P: associated to F1 -> Accurate to ±1 km in 5000 km
  - NF-S(P): nominal speed 800 km/h
  - NF-I: associated to F4 -> Philips A/C 1267 VHF radio

Sub-system Level Requirements



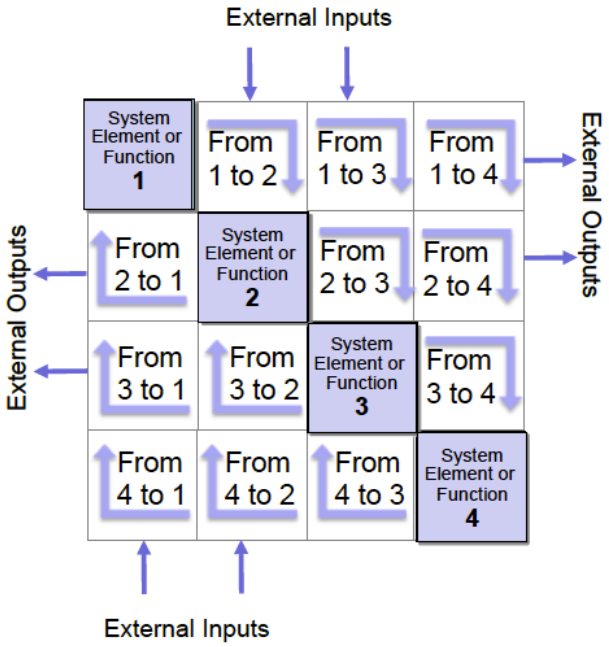
Holistic approach

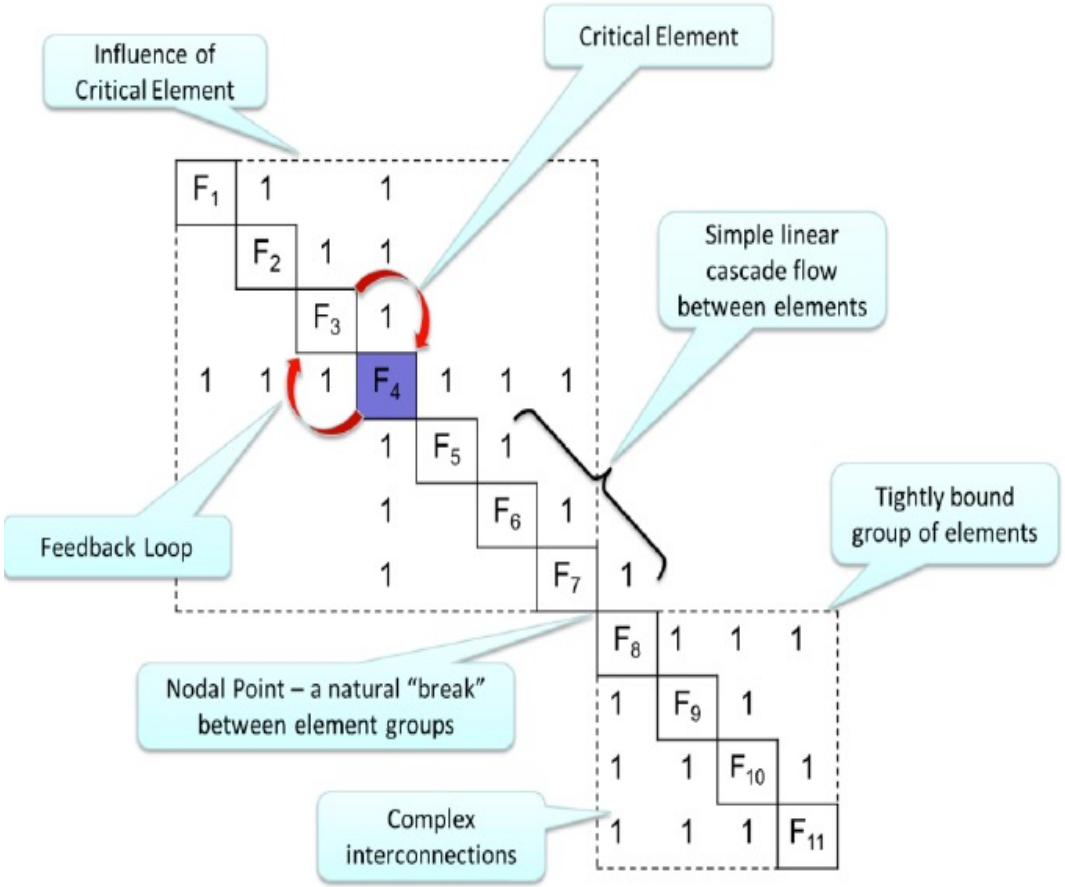
Civil Aircraft example (cont'd):

- System Level:
  - F: Propulsion (F5)
  - NF-P: Consumption < X (A380 3.27 L/100 Km per seat, take-off 230 MW)
- Sub-system Level (F5):
  - System F5 -> Sub-S O
  - System NF-P -> Sub-S NF-S(P) Consumption < X

N^2 analysis

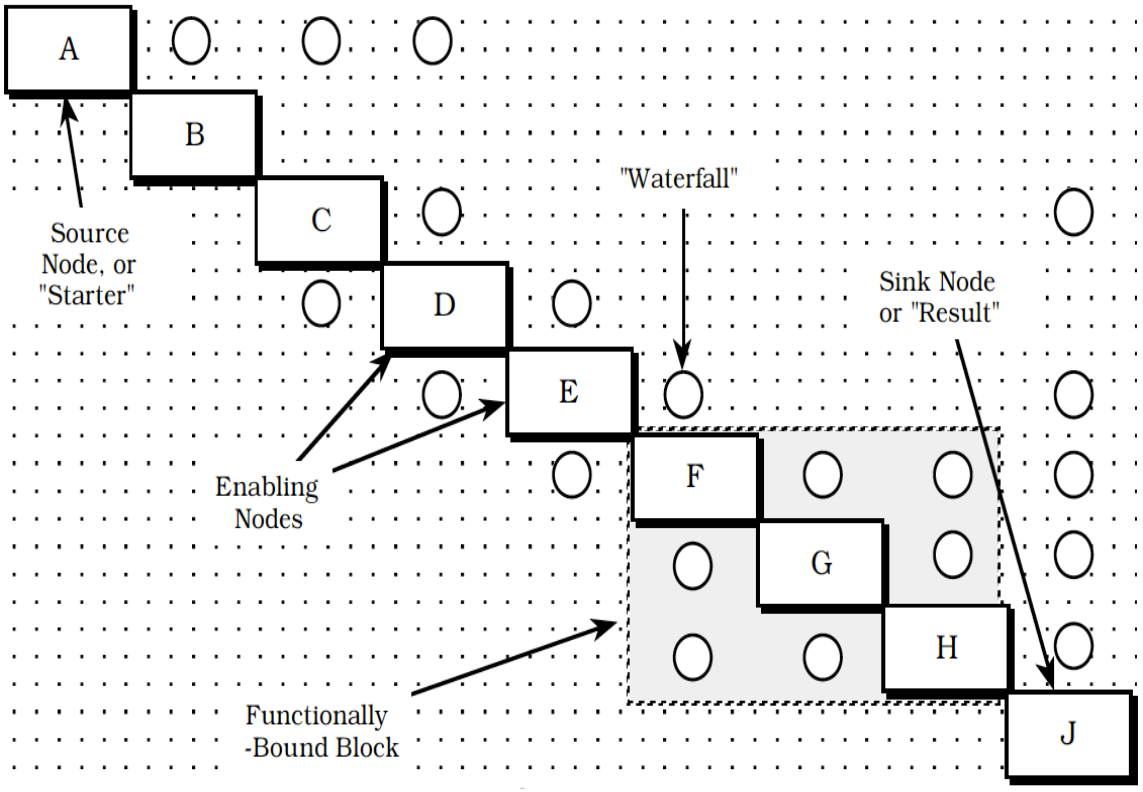
Burge'11

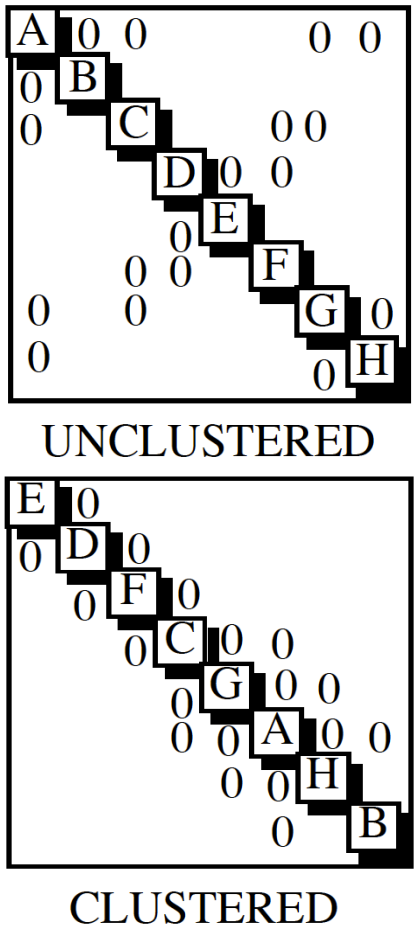




N^2 analysis

Hitchins'93



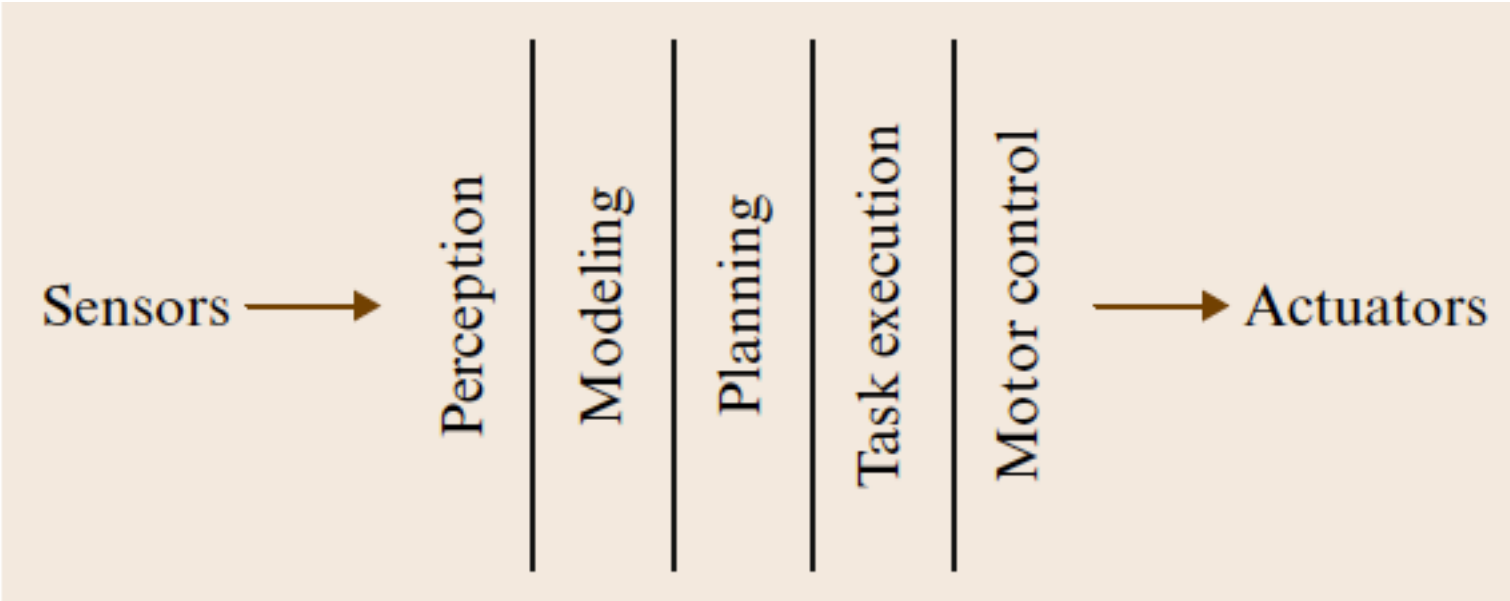


Robotic Systems Architecture

- Communication:
  - Client-server: mesages flow ordered
  - Publish-subscribe: asynchronous messages
- High modularity and Hierarchy:
  - Decrease complexity & increase reliability
  - Multilayer in time, task or spatial abstraction
- Types:
  - Sense-plan-act (SPA)
  - Behavioral (reactive): HL can override LL behaviors
  - Tier/layer-ed

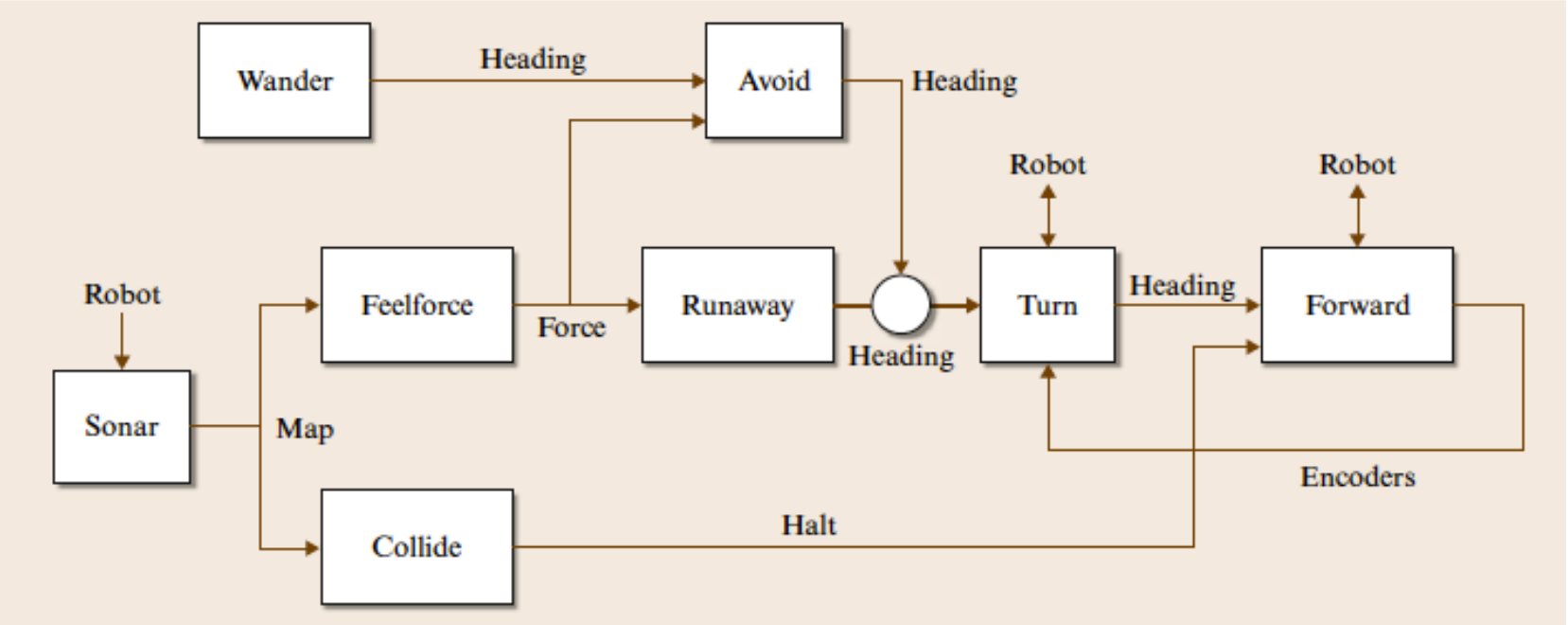
Robotic Systems Architecture

@ SPA (Brooks'86):



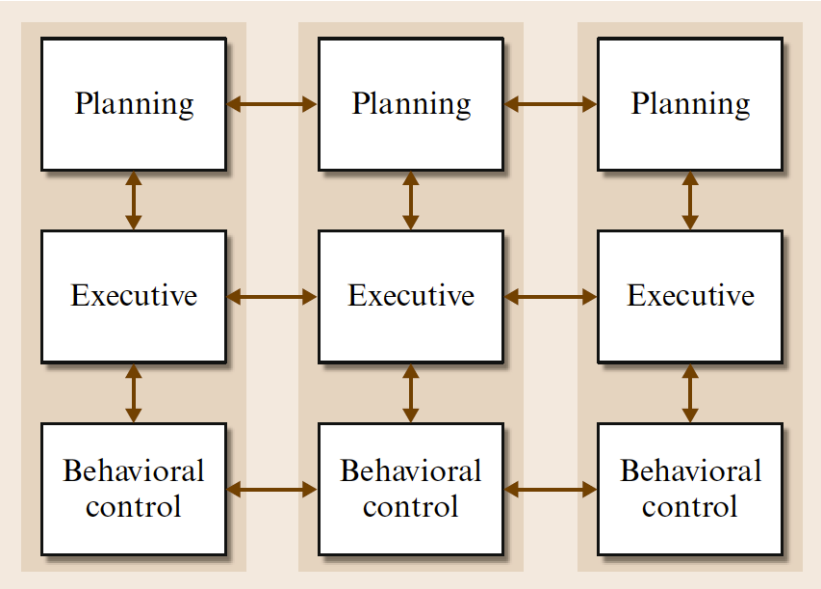
Robotic Systems Architecture

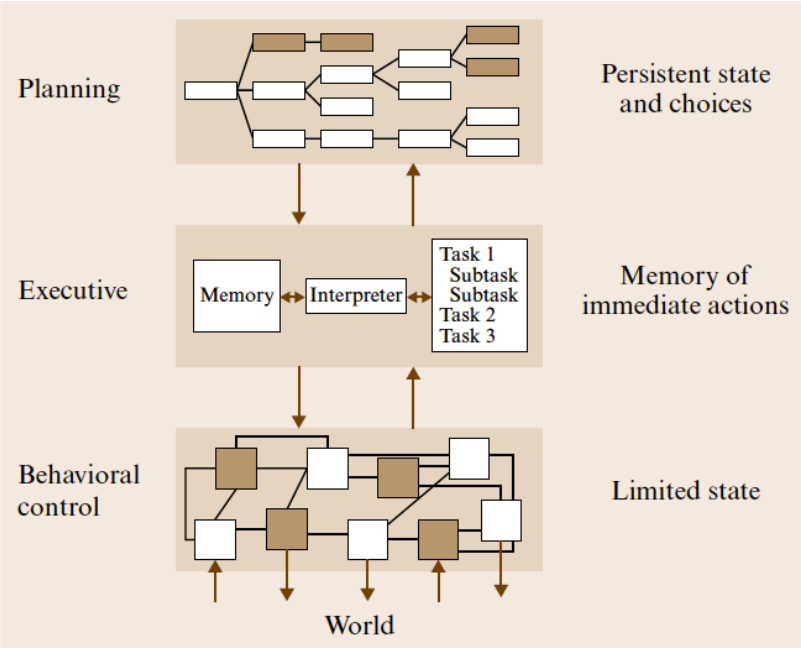
@ Behavioral (Brooks'86):



Robotic Systems Architecture

@ Behavioral (Firby'89): most popular 3T





@ Spacecraft (4T): behav. / exec. / plan. / recovery

Robotic Systems Architecture

3T example Office Delivery Robot:

- Behavioral layer: moving around rooms and halls avoiding obstacles, opening doors
- Executive layer: coordinates the BL for leaving a room, going to an office
- Task-planning layer: decide the order of deliveries to minimize time, sending tasks to executive layer (exit room, go to office)

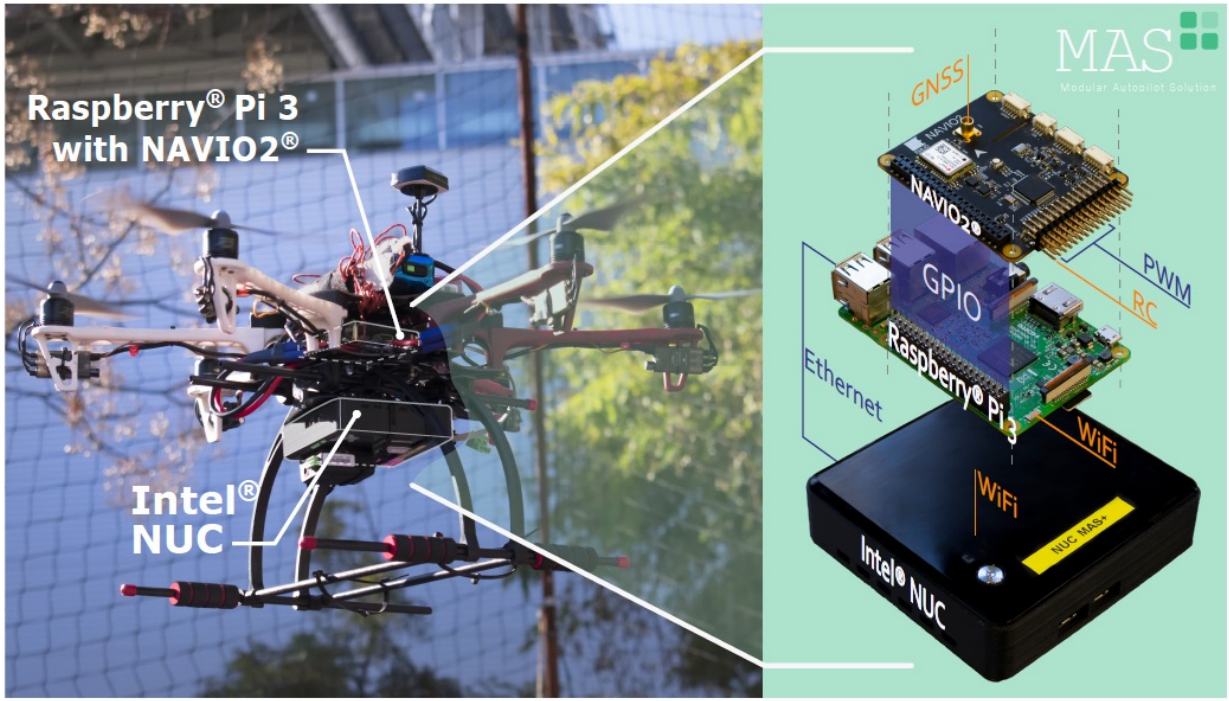
Drones Architecture

PX4 (https://px4.io/):

- Middelware on top of any host OS POSIX (NuttX, Linux) (2T -> 3T): internal/external communications and hardware integration
- Flight stack: estimation and flight control system modules (guidance, navigation and control algorithms)
- Reactive: communication done by asynchronous message passing uORB API, bridges MAVLink, Real Time Publish Subscribe (RTPS)

Drones Architecture

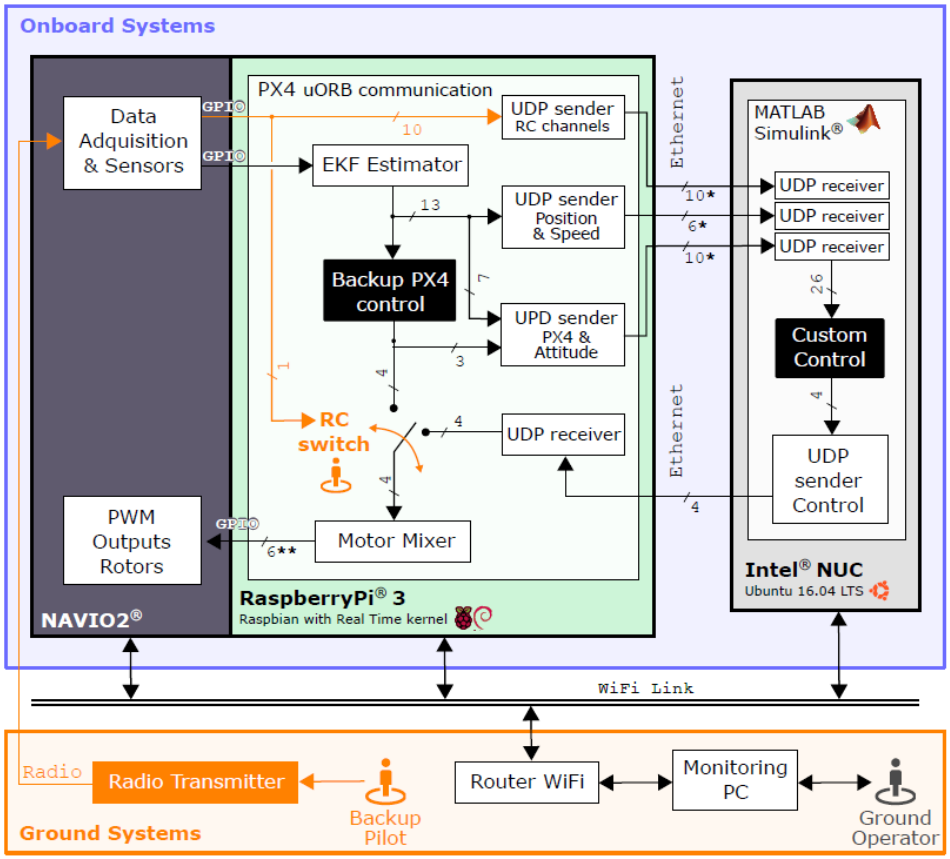
MAS+ Hardware (prototyping)



https://ieeexplore.ieee.org/document/9291382

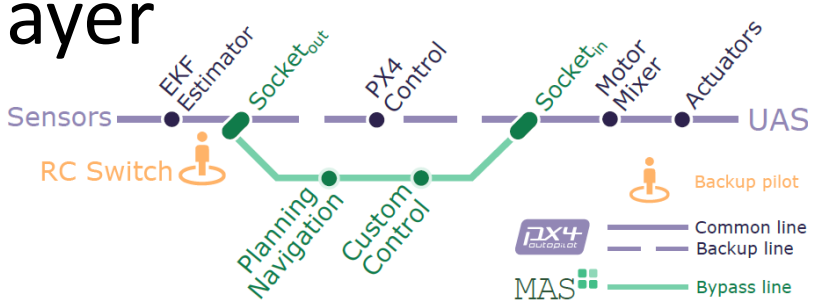
Drones Architecture

MAS+ Hard/Soft-ware 3T Architecture



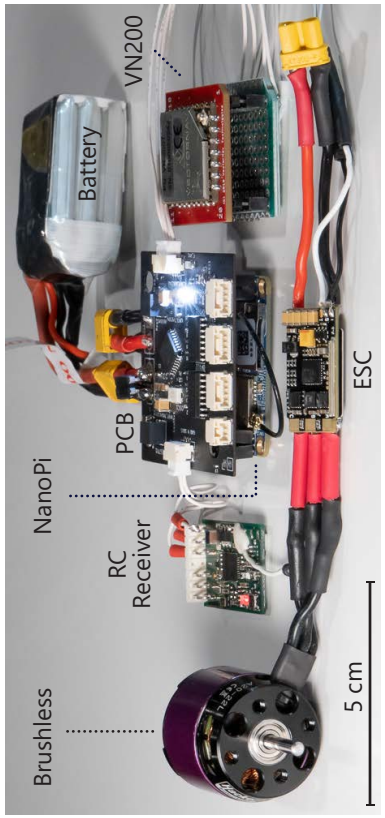
Drones Architecture

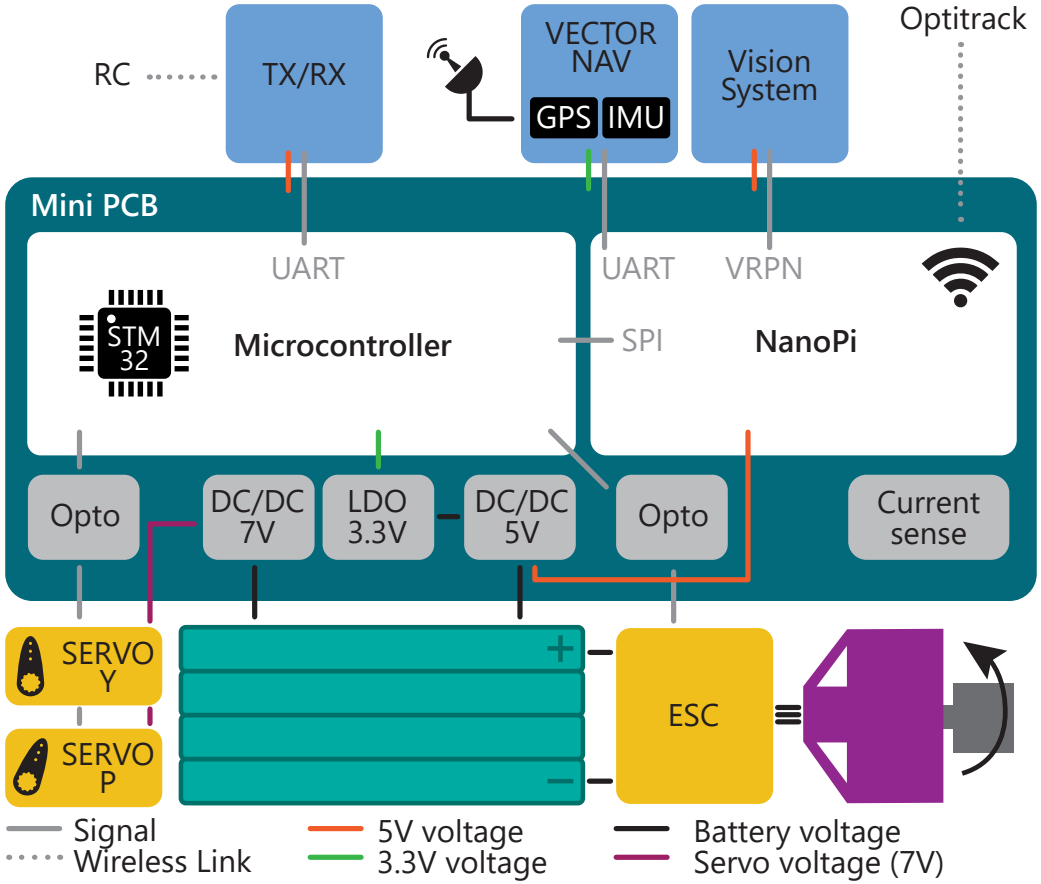
MAS+ Control/Planning layer



Drones Architecture

SPA Architecture:





http://ras.papercept.net/images/temp/IROS/files/1994.pdf