

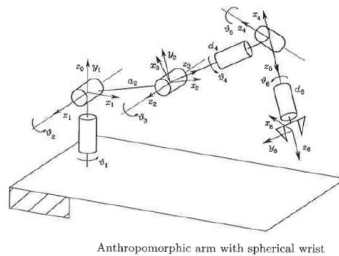
# An Architect's Agile Journey

Insights and Lessons from Medtronic's Surgical Robotics  
Division

# Luca Ingianni and Tom Baker

## Tom Baker

Mechanical Engineer  
(Robotics, Dynamics,  
and Controls)



Robotics Researcher

## Medtronic

- Controls Software Engineer
- Robot Behavior Software Architect

## Luca Ingianni

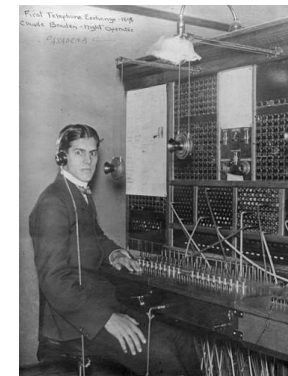
Aeronautical Engineer



Embedded systems  
jack-of-all-trades

Co-founder of vempio:  
agile coach, trainer and  
explainer

Vocabularius rerum



Podcasting Addict:

- Agile Embedded Podcast
- DevOps auf die Ohren und ins Hirn
- LEAP: value-driven leadership

# Medtronic Hugo™ robotic-assisted surgery (RAS) system







# 74M+

In the past year, Medtronic therapies improved the lives of 74 million+ people. That's **two people every second** of every hour of every day – and counting.

## Engineering the extraordinary

150+  
countries

95K+  
employees

78  
manufacturing  
sites

40  
labs and research  
development sites

214  
active clinical trials  
last year

\$2.7B  
in R&D investments  
last year

12.4K+  
scientists  
and engineers

46K+  
active  
patent matters

Our dedication to transforming lives starts here

Better  
outcomes for  
our world

Life-transforming  
technologies

Experiences  
that put  
people first

Insight-driven  
care

### Beyond products: Integrated health solutions (IHS)

We partner with hospital management and medical leaders to transform care pathways and clinical operations, with a shared goal of:

- Improving efficiency and outcomes
- Increasing patient and staff satisfaction
- Optimizing cost

With our extensive experience across and within regions, we provide proven and highly effective solutions that can be quickly tailored to providers' specific needs and generate value now – and for the future.

In 2023, IHS celebrates a decade of transforming health-care, delivering value for patients and healthcare professionals, and improving access to high-quality care while optimizing costs and capacity.

### Positive impact on complex and challenging conditions

70+

**conditions** in the human body  
treated with our therapies

2

**people every second**  
have their lives improved –  
and counting

### Public-private partnerships

Increasing patient access in multiple communities aligns with the core of our Mission. These public private partnerships – with outcome-oriented business models – create mutual benefits by making the most of Medtronic skills, expertise, and assets through the appropriate transfer of risk. They drive our globalization strategy and actively contribute to the international push for universal health coverage. With jointly defined objectives, they address health system challenges efficiently by increasing access to therapy innovation, raising quality, and delivering improved outcomes.

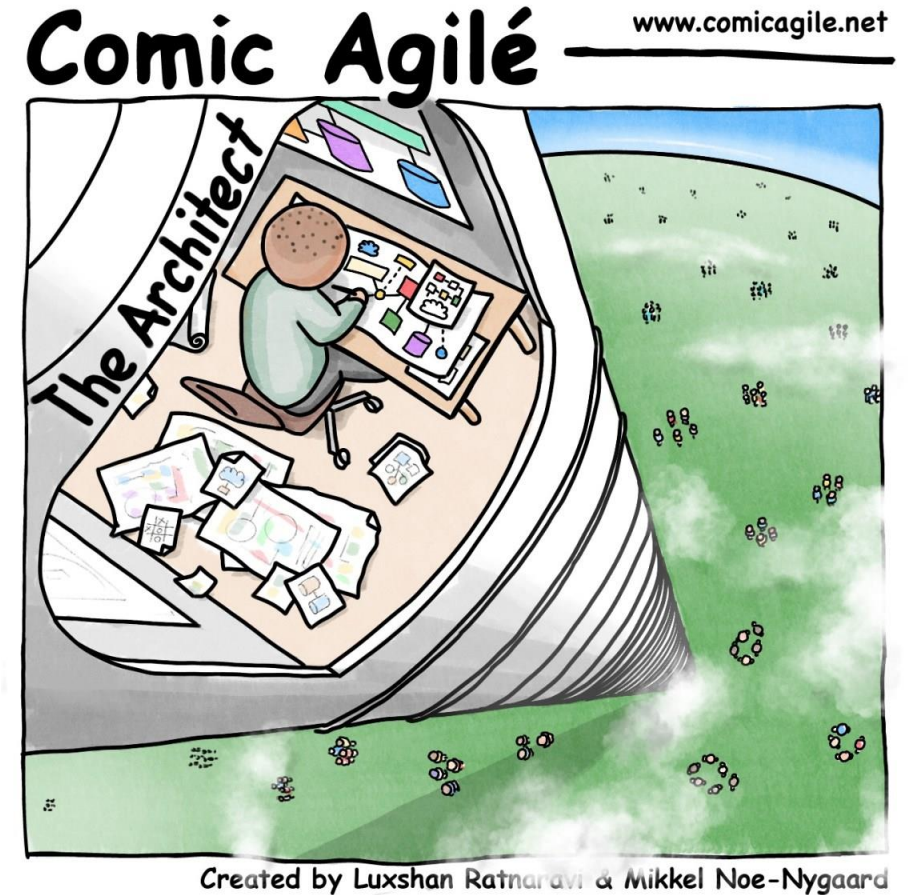
Learn more at [Medtronic.com](https://www.Medtronic.com)

©2023 Medtronic. Medtronic, Medtronic logo, and Engineering the extraordinary are trademarks of Medtronic. All other brands are trademarks of a Medtronic company. UC202206905b EN

**Medtronic**  
Engineering the extraordinary

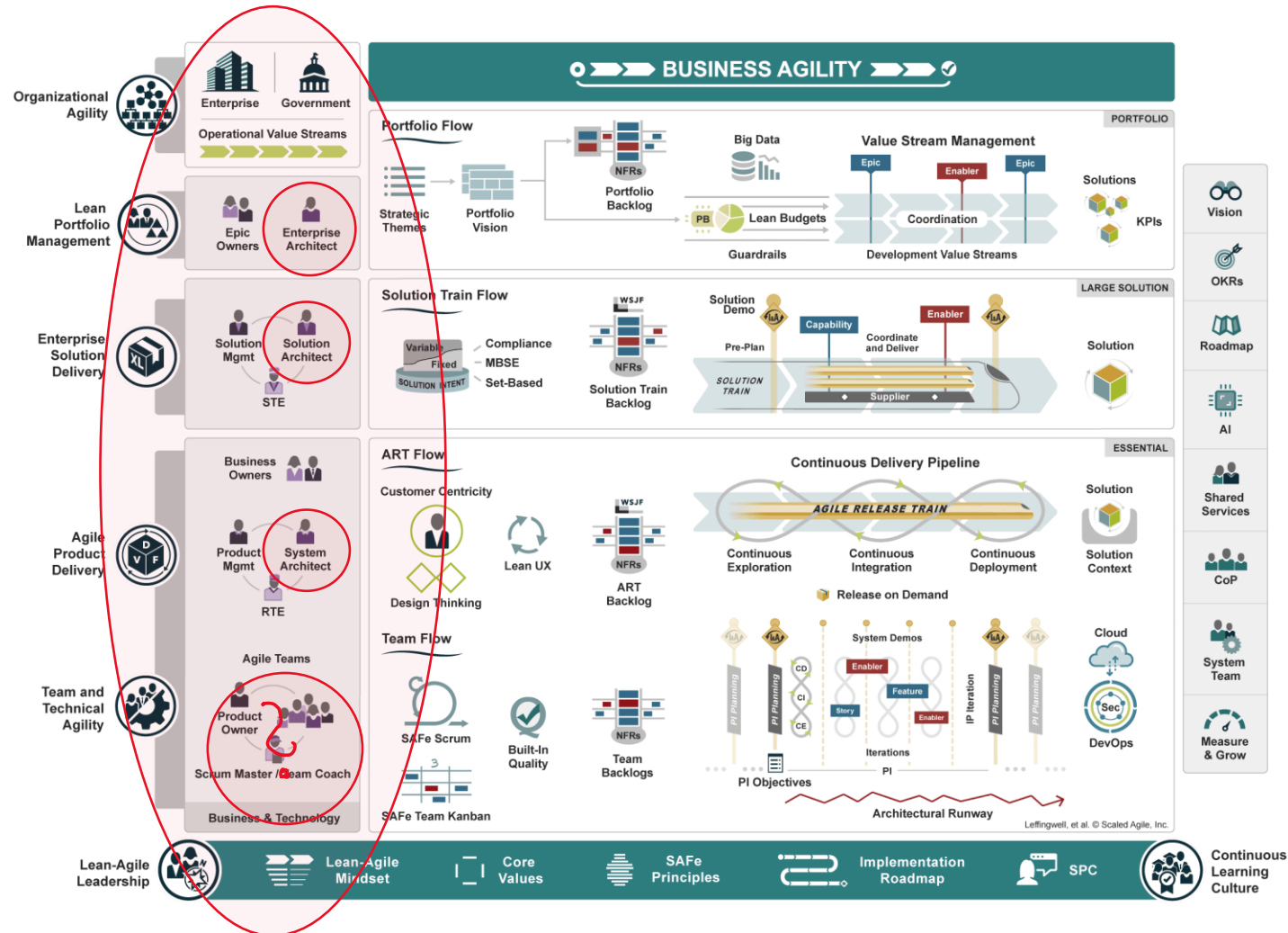
# What does a Software Architect do?

- Origin of the word the term is Greek: arkhi-tektōn  
--> chief builder
- "Super"-senior developer?
- Develops software?
- Abstracts the software?
- Draws boxes and arrows?
- Constraints the developer's creativity?
- Focuses on the technology behind the product?
- Schemes (dreams) the vision?
- Sits in their ivory tower?
- Works in their own development environment?





# Where we started: SAFe's Idea of Architects

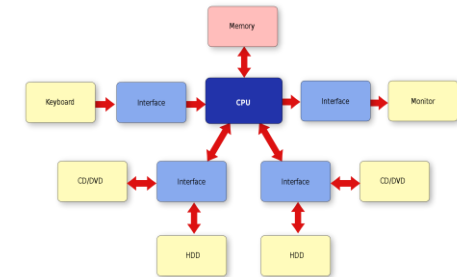


# What's the Place of an Architect?

- High level: Product/requirement considerations
  - Interfacing product management and the business?



- Mid level: "Architect" the system, the famous „boxes and arrows“



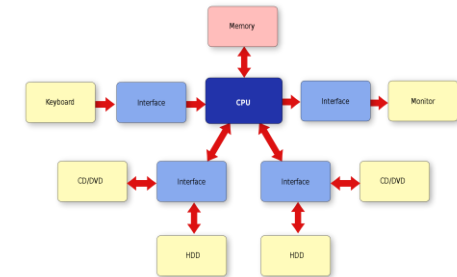
- Low level: Write code, tests or other things having to do with curly brackets
  - Interfacing software developers?

```
/**
 * Simple HelloButton() method.
 * @version 1.0
 * @author john doe <doe.j@example.com>
 */
HelloButton()
{
    JButton hello = new JButton( "Hello, wor
    hello.addActionListener( new HelloBtnList

    // use the JFrame type until support for t
    // new component is finished
    JFrame frame = new JFrame( "Hello Button"
    Container pane = frame.getContentPane();
    pane.add( hello );
    frame.pack();
    frame.show();           // display the fra
}
```

# Tensions and Struggles of an Architect

- High level: Product/requirement considerations
  - Interfacing product management and the business?
  - No time for "nice-to-have" re-factoring
  - New features are more important
- Mid level: "Architect" the system, the famous „boxes and arrows“
- Low level: Write code, tests or other things having to do with curly brackets
  - Interfacing software developers?
  - Tension with developers (stepping on their toes)
  - Bottleneck – One architect supporting multiple teams



```
/**
 * Simple HelloButton() method.
 * @version 1.0
 * @author john doe <doe.j@example.com>
 */
HelloButton()
{
    JButton hello = new JButton( "Hello, wor
    hello.addActionListener( new HelloBtnList

    // use the JFrame type until support for t
    // new component is finished
    JFrame frame = new JFrame( "Hello Button"
    Container pane = frame.getContentPane();
    pane.add( hello );
    frame.pack();
    frame.show();           // display the fra
}
```



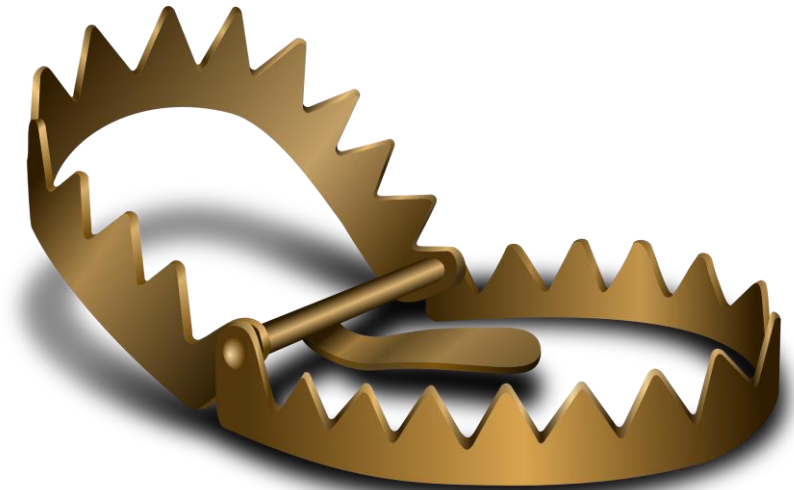
# The „Golden Middle“ is a Trap

## **Retreats to their ivory tower:**

- Works in isolation
- Disengaged from the day-to-day
- Loses connection to business and developers

## **Focus on drawing boxes and arrows:**

- A lot of effort to create good documentation
- Who actually reads the documentation?
- Boxes and arrows by themselves aren't helpful for anyone



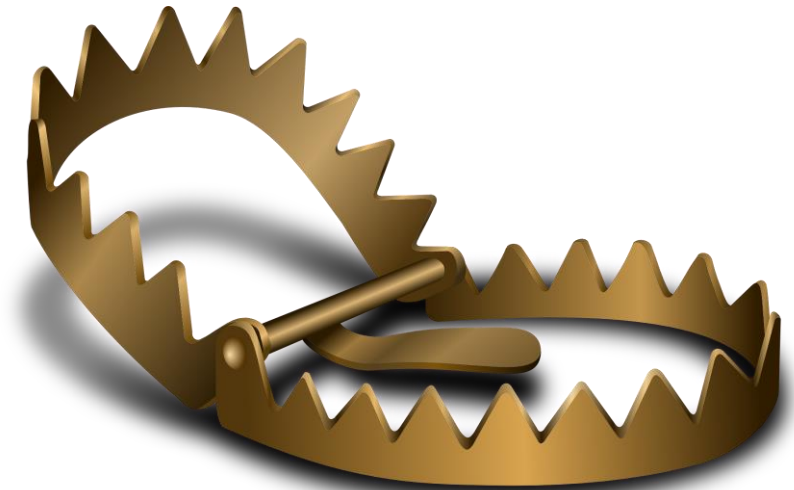
# The „Golden Middle“ is a Trap

## **Retreats to their ivory tower:**

- Works in isolation
- Disengaged from the day-to-day
- Loses connection to business and developers

## **Focus on drawing boxes and arrows:**

- A lot of effort to create good documentation
- Who actually reads the documentation?
- Boxes and arrows by themselves aren't helpful for anyone



➤ How can the trap be avoided?

# Interfacing Product Management/Business

- Need the **trust and support** of the business
  - State the technical vision in non-technical terms
  - Relate to business needs
- Focus on the **value**
  - Continuous delivery: Develop an architecture that allows change in small steps
  - Always prioritize focus on improvements that serve the business needs

# Interfacing Software Developers

- Ensure that the **vision and goal** is known and understood
  - Beware of over-constraining --> every constraint must have a good reason
- Define **clear boundaries** of responsibilities
  - For example: The interaction of components are the architects' responsibility, the components are the developers' responsibility
- **Communication over documentation**
  - Speak the same language (Focus the architecture description on concrete components and interfaces)



# The Architect as the Connection between the two Sides

- **Translate inputs from product level** to something more easily digestible by developers (even header files perhaps?)
- Be the **voice of developers** (and „the system“) towards product management/business
- Create a **shared understanding** (even a shared language)
- Insulate against **cost of change**



# What about the Boxes and Arrows?

- Documentation is crucial in regulated industries, it is an integral part of the product
- It's part of the architect's responsibility to ensure correct, timely, complete, traceable documentation
- Use tools and methods that connect both sides
  - Connects to requirements, etc.
  - Connects to code (part of the repository)

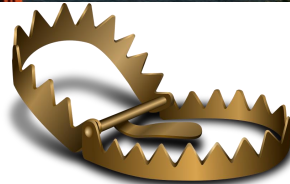


- **Documentation as Part of the Product**
- **Communication is key, if boxes and arrows support that, then use them**

# Main Learnings

A Software Architect is not the super-senior, white-haired chief developer, instead:

- Connects with developers and product management, each on their level
- Serves as the conduit between development and product management
- Has a holistic view of the product, so others are free to pursue their own niches
- Needs to do **continuous work** not to accidentally be “forced into the tower”



```
/**
 * Simple HelloButton() method.
 * @version 1.0
 * @author john doe <doe.j@example.com>
 */
HelloButton()
{
    JButton hello = new JButton( "Hello, wor
    hello.addActionListener( new HelloBtnList

    // use the JFrame type until support for t
    // new component is finished
    JFrame frame = new JFrame( "Hello Button"
    Container pane = frame.getContentPane();
    pane.add( hello );
    frame.pack();
    frame.show();           // display the fra
}
```

# What we're still Figuring out

- How to best fulfill the architect's position and role, especially in our medical device context
- How to create good, valuable documentation
  - For the developers
  - For the product managers
  - For the regulatory body
- How to go towards Documentation as Code and Documentation as part of the Product



# Thank You!

