ETEL MOTION TECHNOLOGY

Innovative Motion Control

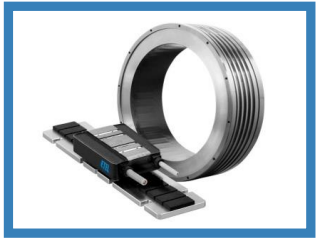
MATLAB Helps Save Time and Effort by Streamlined Commissioning of the QuiET System

Ralph Coleman
Senior Researcher
Advanced development

23.06.2016

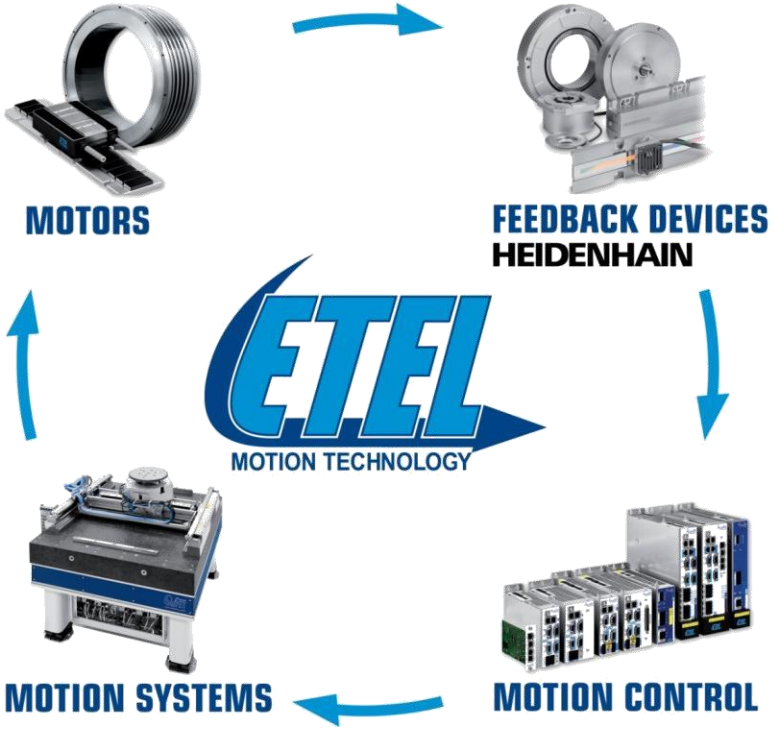
Outline

- Company presentation
- Product development workflow
- Active isolation systems
- Commissioning tool
- Results
- Conclusion



ETEL product range

A comprehensive motion control solution



The company

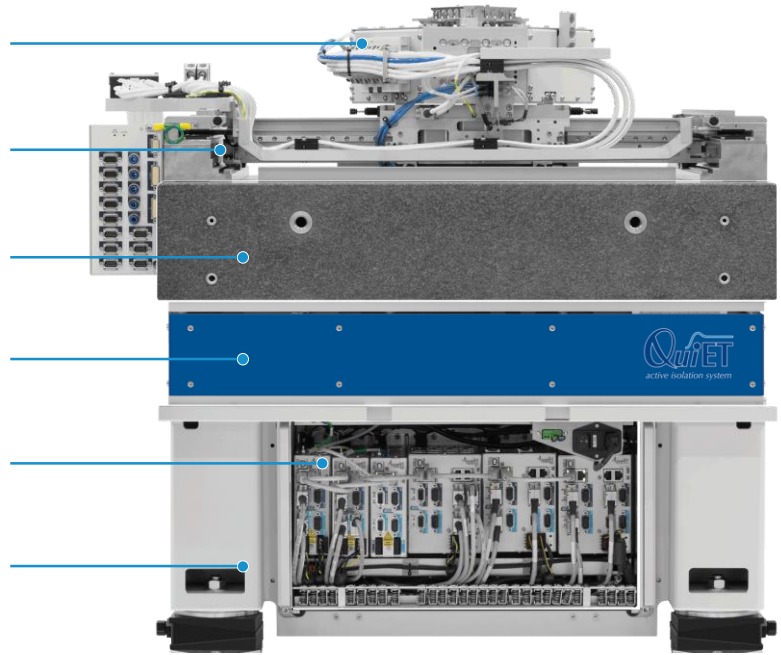
General data 2015

- **Headquarters:** Môtiers / Switzerland
- **Capital:** 17.8 million CHF
- **Net sales:** approx. 60 million CHF
- **Building area:**
  - Môtiers – 14’700 m^2
  - Couvet – 1’660 m^2
- **Worldwide employees:** 349

ETEL forward integration

From components to advanced motion systems

- MOTION SYSTEM
- MOTORS
- BASE
- ACTIVE ISOLATION SYSTEM
- MOTION CONTROL
- FRAME



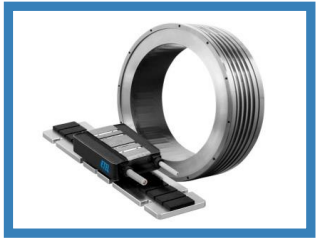
Key takeaways

- **Reduce the development time of control algorithms by around 40% by using high-level programming in MATLAB and Simulink**

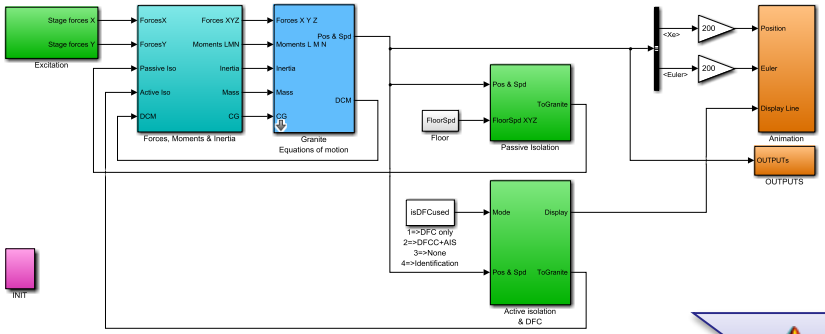
- **Software tools:**
  - Developed simultaneously by using the same interface with Simulink Real-Time & stand-alone products
  - Deployed with small changes only to avoid error-prone implementation

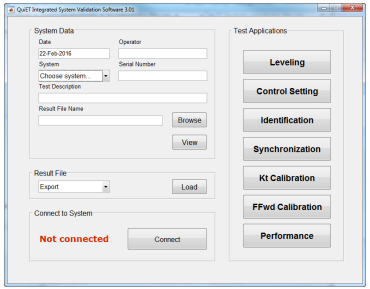
Outline

- Company presentation
- **Product development workflow**
- Active isolation systems
- Commissioning tool
- Results
- Conclusion



Product development workflow





Simulation

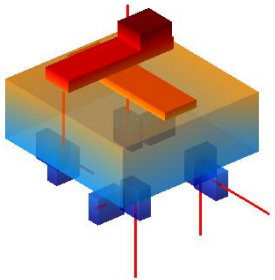
Commissioning tool
(set-up and tuning of parameters)

Deploy

HIL simulation

Stand-alone

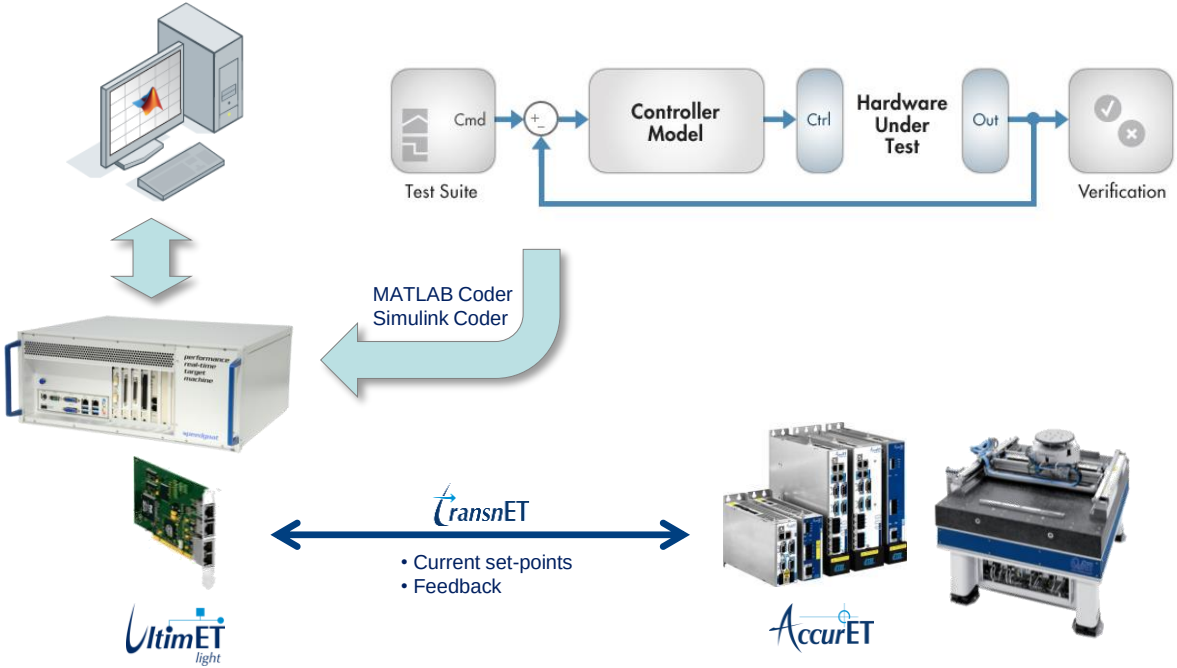
ANALOG DEVICES
XILINX



HIL for development of control & commissioning

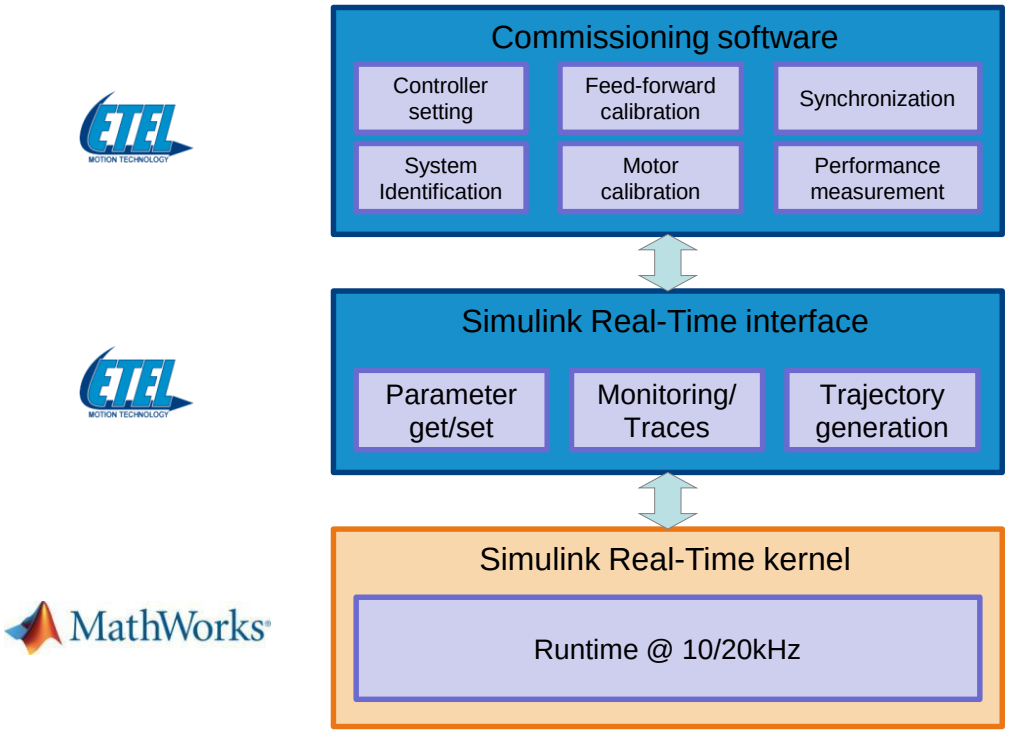
Simulink Real-Time used to accelerate development

- Controller in Simulink
- Commissioning software in MATLAB OOP



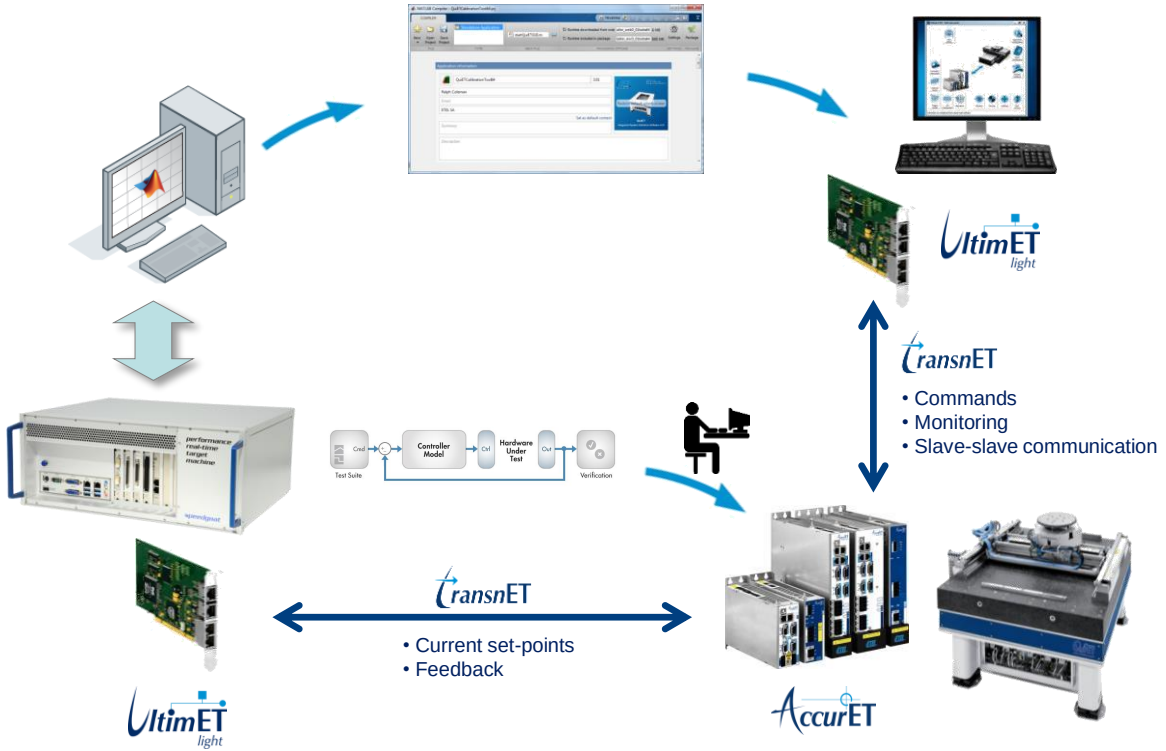
Software structure during development

Interface with Simulink Real-Time



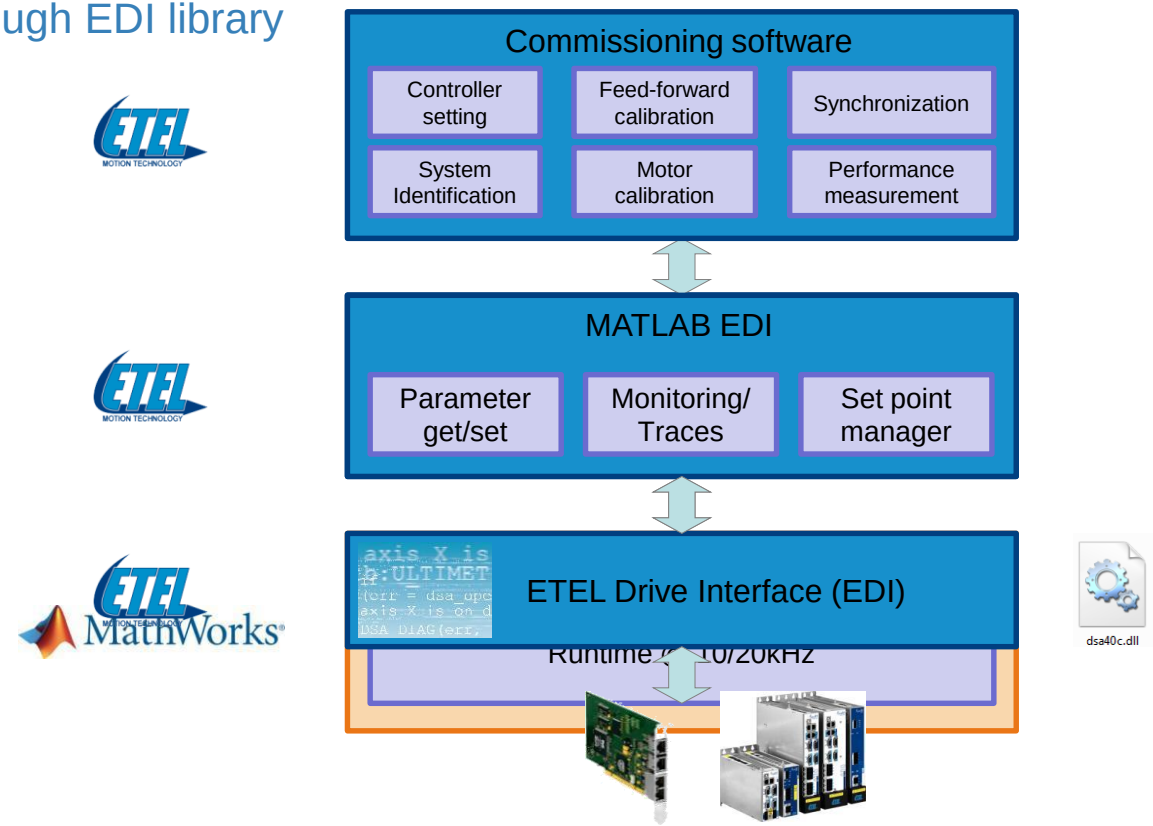
Switch over to stand-alone system

MATLAB Compiler used to create stand-alone application



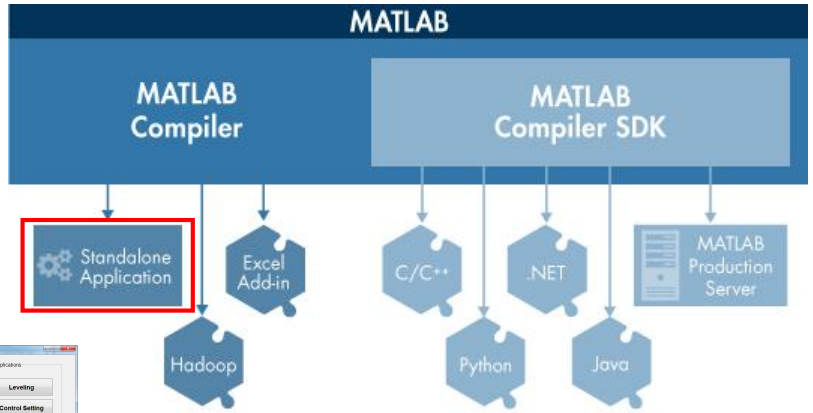
Software structure when deployed

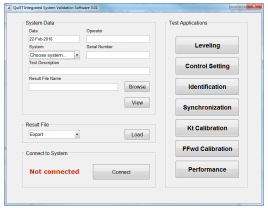
Interface through EDI library



Stand-alone application

Currently investigating the creation of plug-ins for ComET

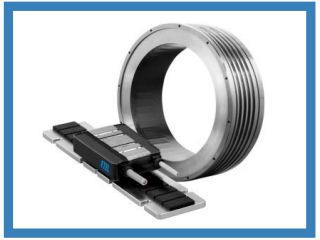




Stand-alone commissioning tool

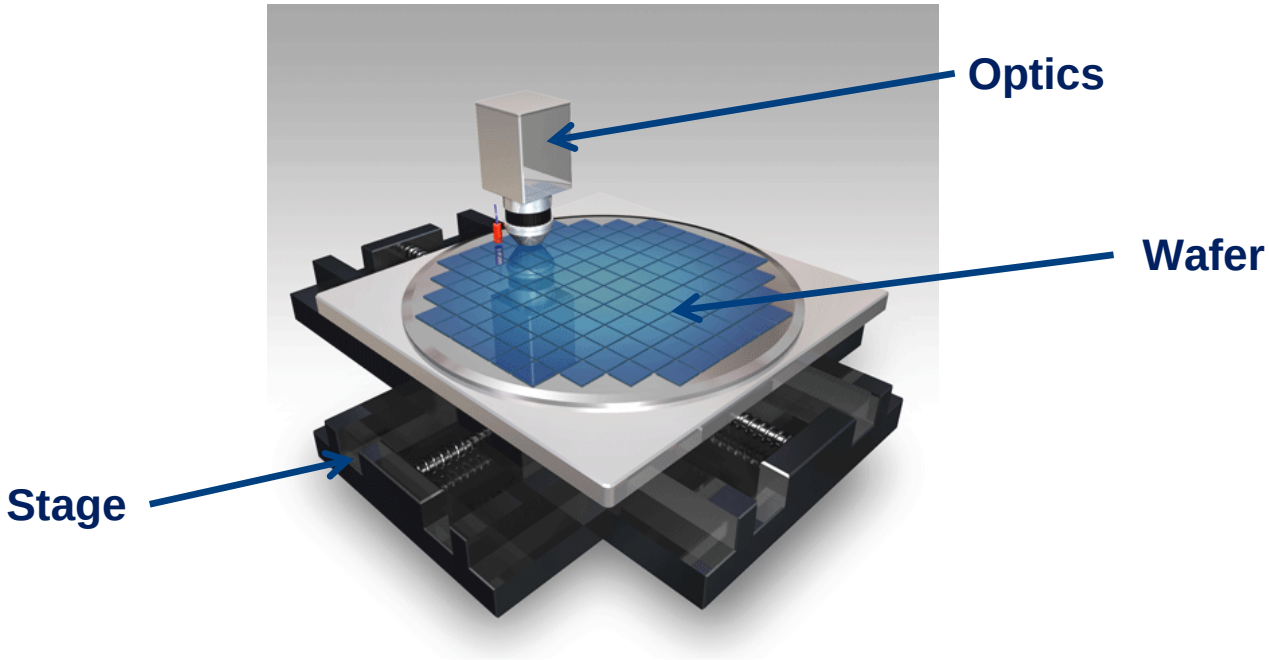
Outline

- Company presentation
- Product development workflow
- **Active isolation systems**
- Commissioning tool
- Results
- Conclusion



Isolation systems

Motivation: Semiconductor market, lithography & process control



The relative position between optics and wafer should be stable to obtain a focused image

Isolation systems

Motivation: Semiconductor market, lithography & process control

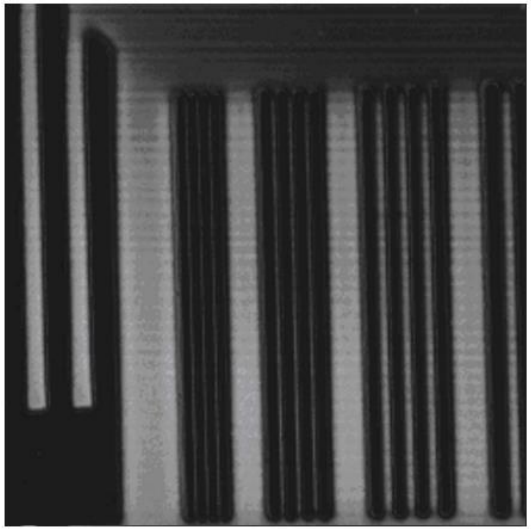


Image with poor position stability

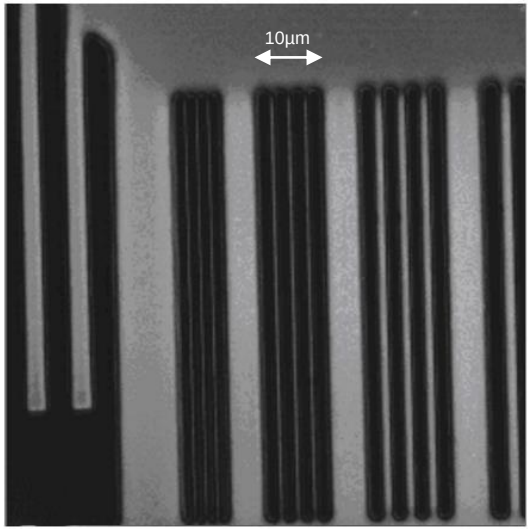
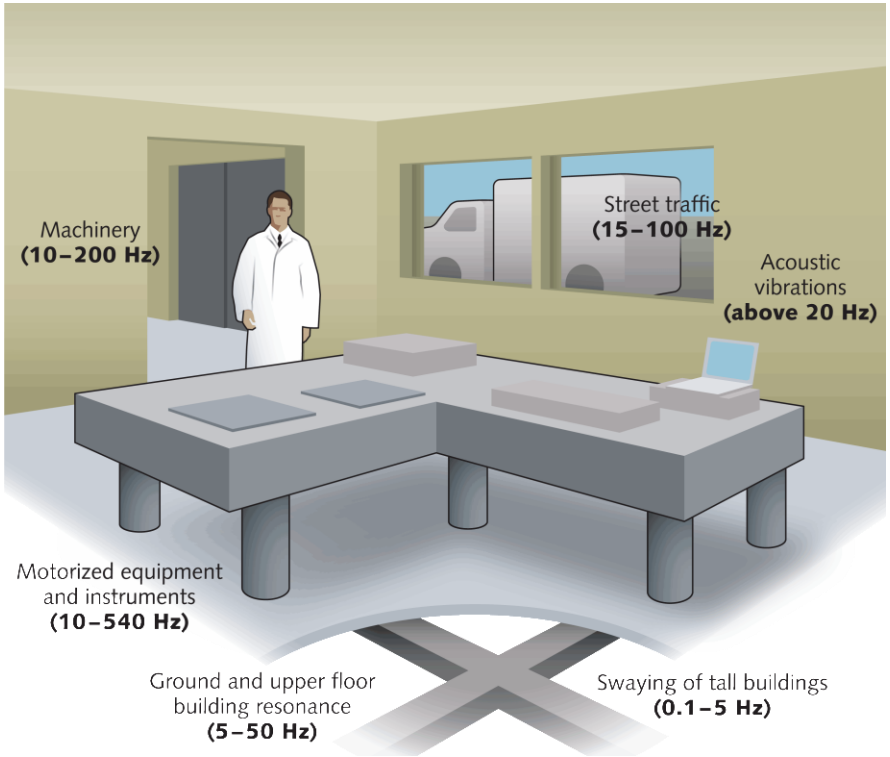


Image with good position stability

Source http://www.hwlscientific.com/news/press.html

Isolation systems

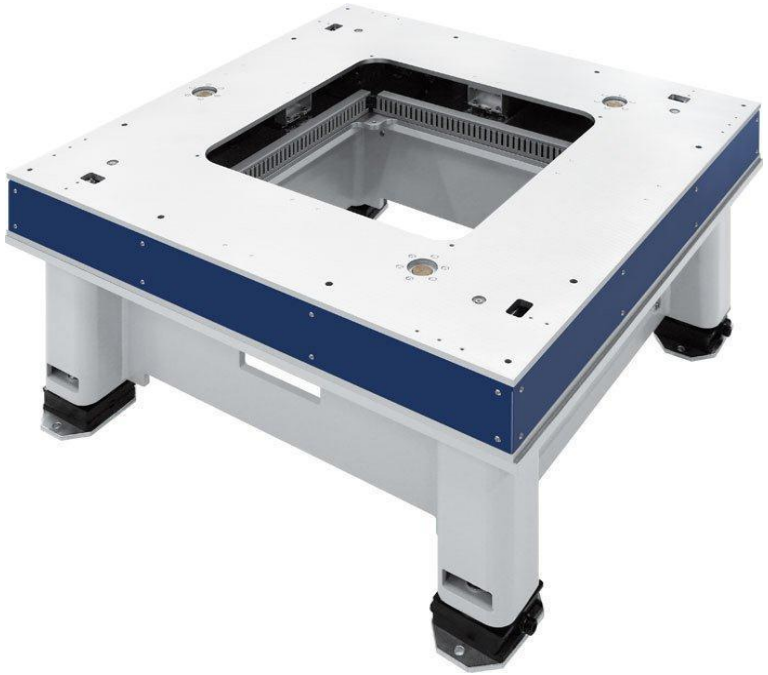
Typical floor vibration sources



Source http://www.laserfocusworld.com/articles/2010/06/vibration-control.html

QuiET

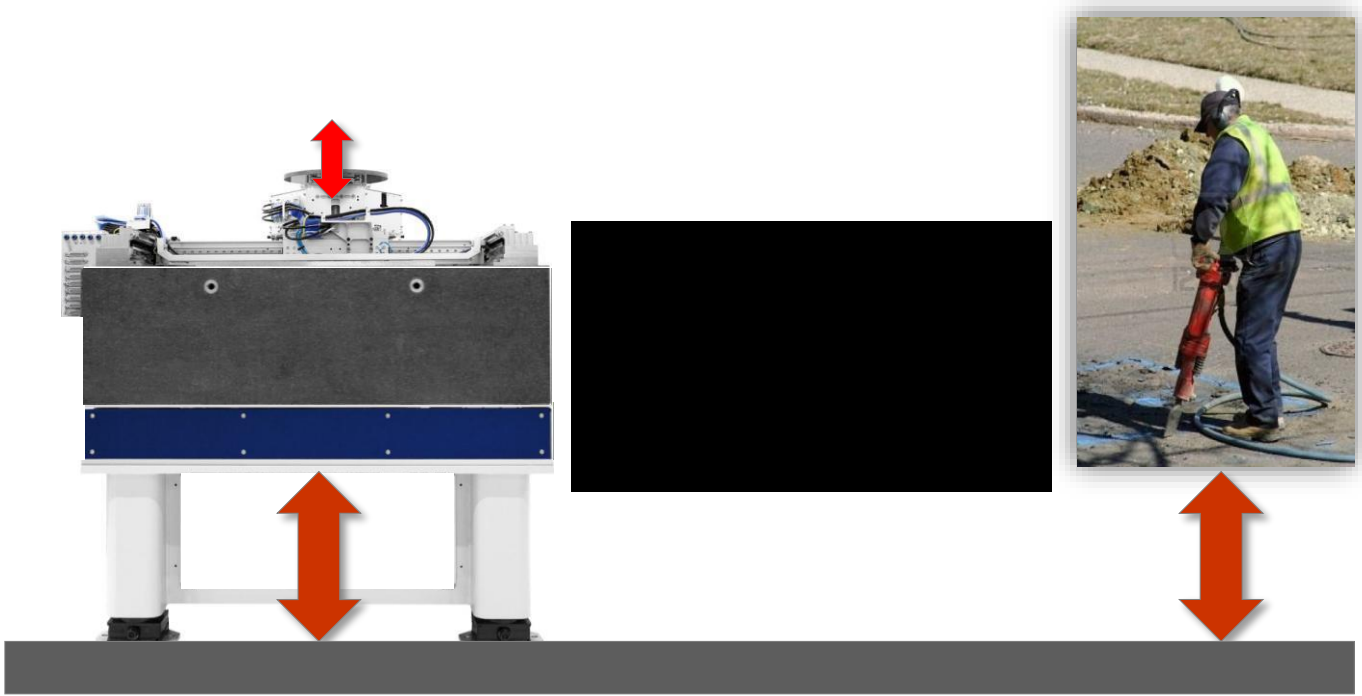
A brief introduction



QuiET
active isolation system

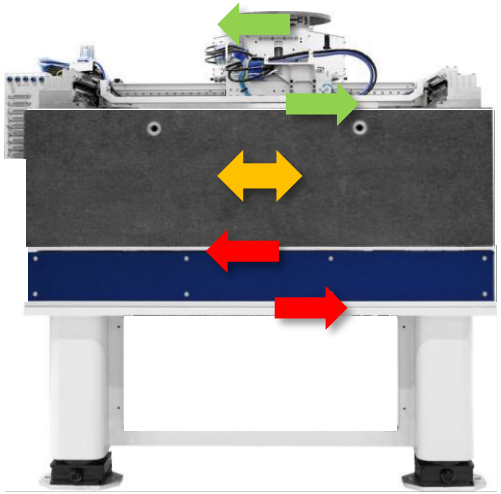
QuiET – Floor vibration isolation

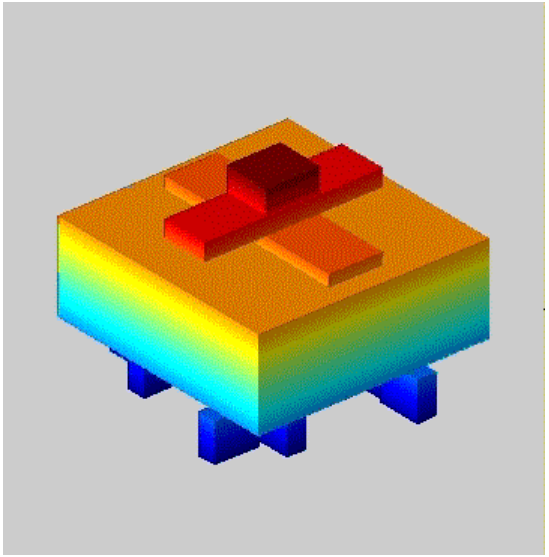
A brief introduction



QuiET – Stage movement compensation

A brief introduction



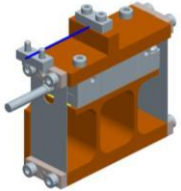


Simulation with badly tuned parameters and base movements magnified x200

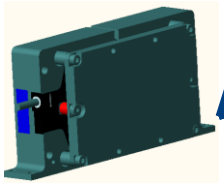
Mechanical assembly

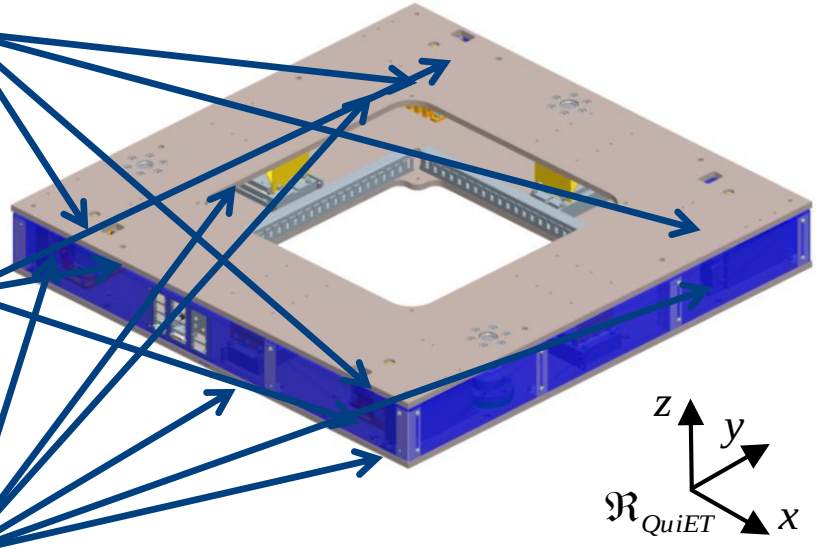
4 isolators

6 sensors
(only 3 in figure)



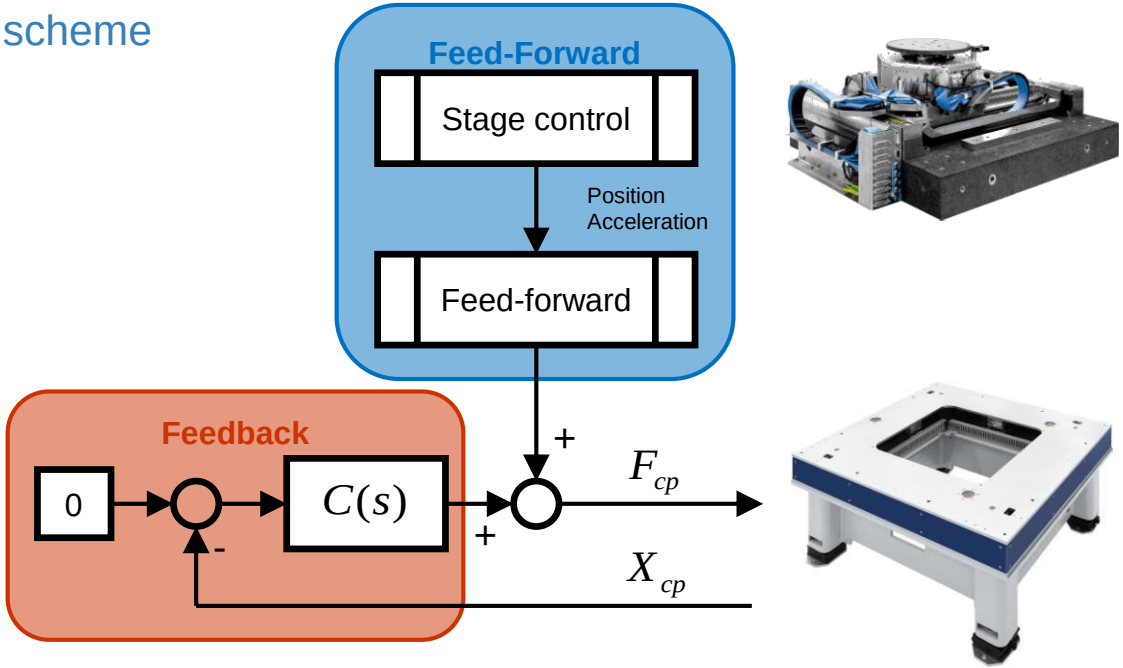
8 motors





System overview

Global control scheme

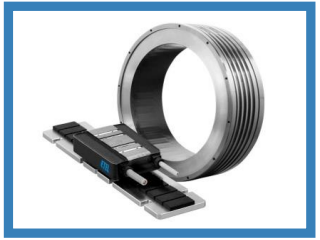


→ **6 Degree-of-Freedom (DoF) systems are tough to get up-and-running**

- All calculations with matrices
- Inter-axis communication issues

Outline

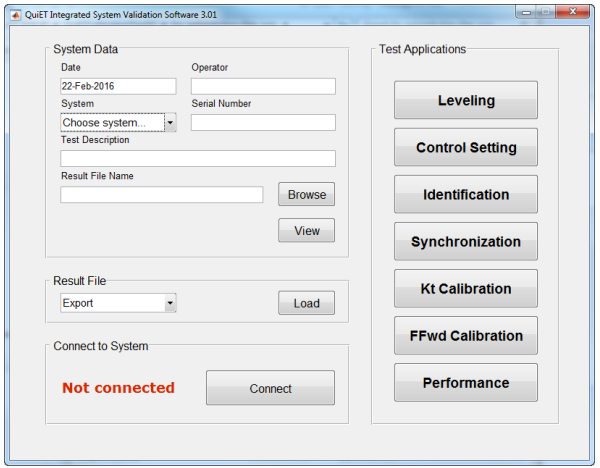
- Company presentation
- Product development workflow
- Active isolation systems
- **Commissioning tool**
- Results
- Conclusion



Commissioning tool general overview

Available features by user

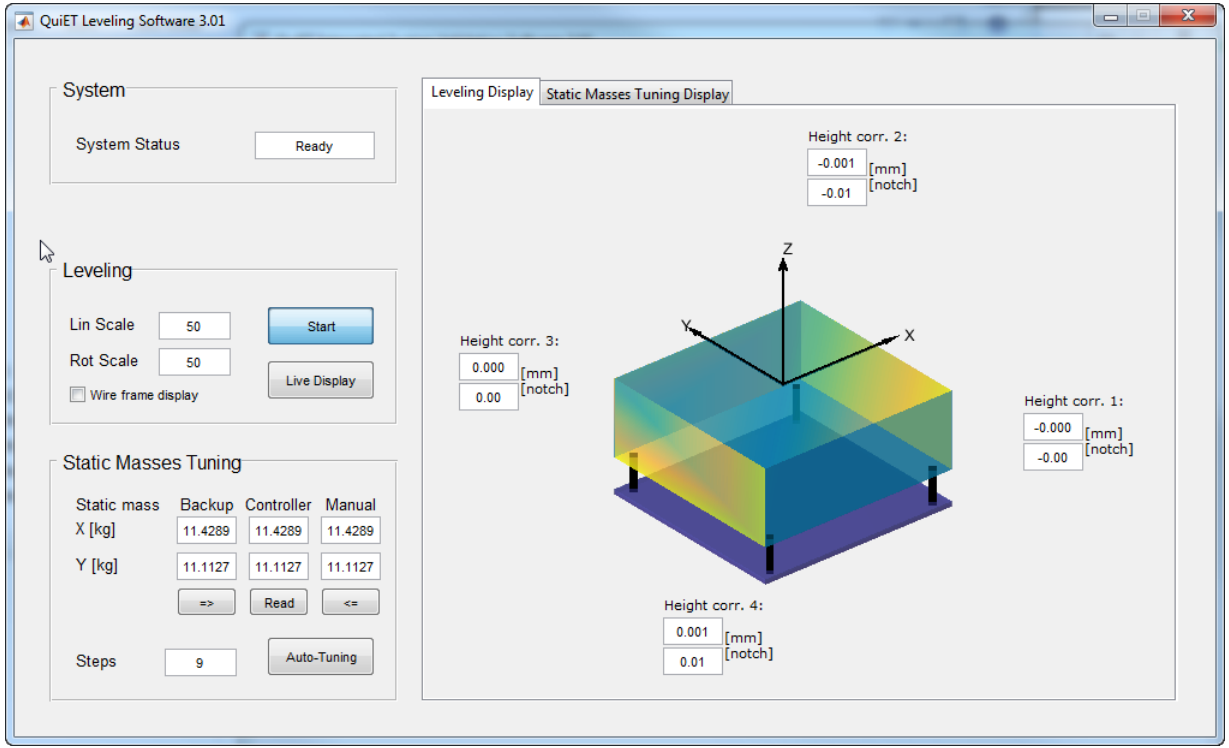
Feature	Customer	Internal use
Leveling	✓	✓
Controller setting	✓	✓
Transmissibility measurement	✓	
System Identification		✓
Synchronization		✓
Motor calibration		✓
Feed-forward calibration	✓	✓
Performance measurement	✓	✓



→ **Deployed as stand-alone application using the MATLAB Compiler for customers and internal use**

QuiET Commissioning Tool

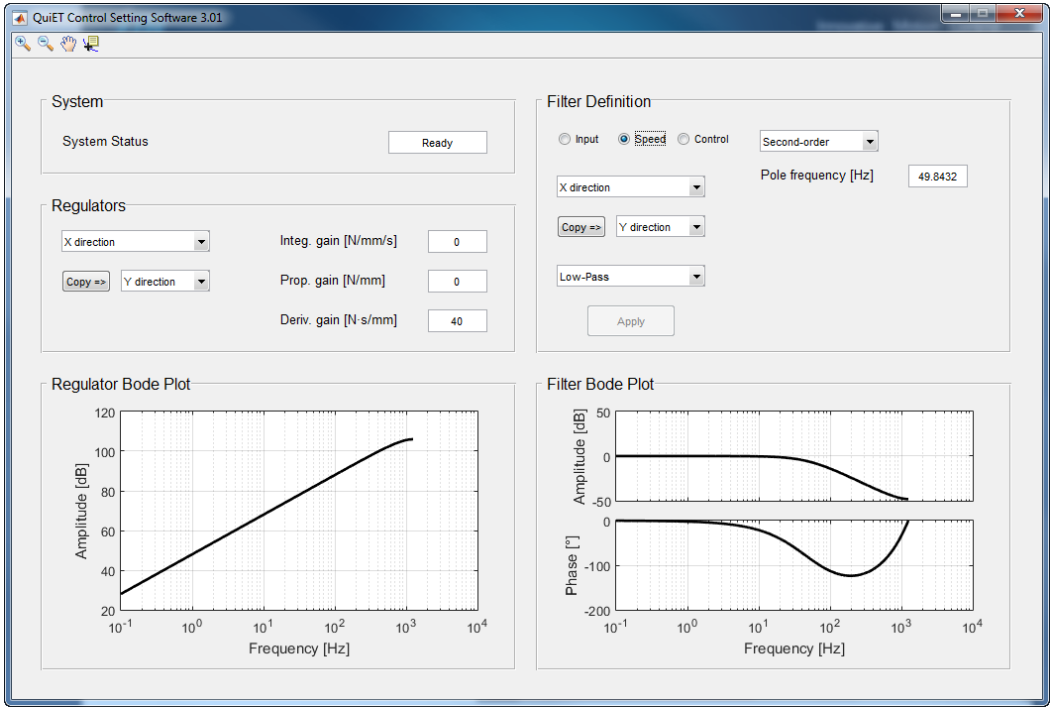
Leveling



QuiET Commissioning Tool

Feedback controller setting

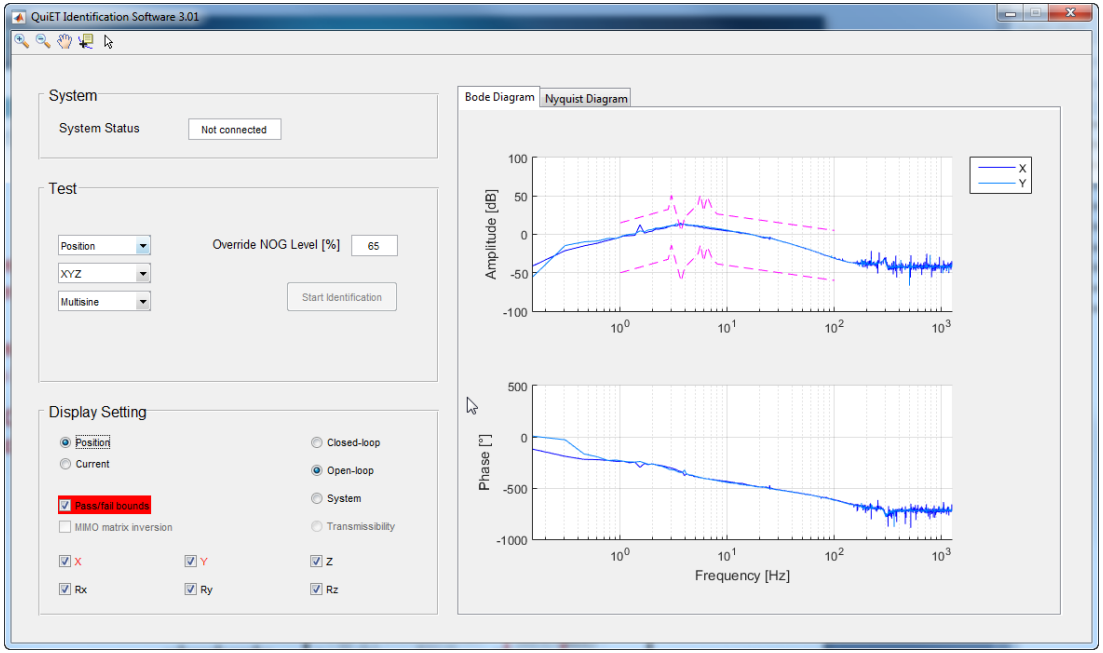
- **This tool is used to set the PID controllers and the filters**



QuiET Commissioning Tool

System identification

- **This tool is used to identify all transfer functions in 6 DoF**
- **Intensive use of the Control System toolbox**

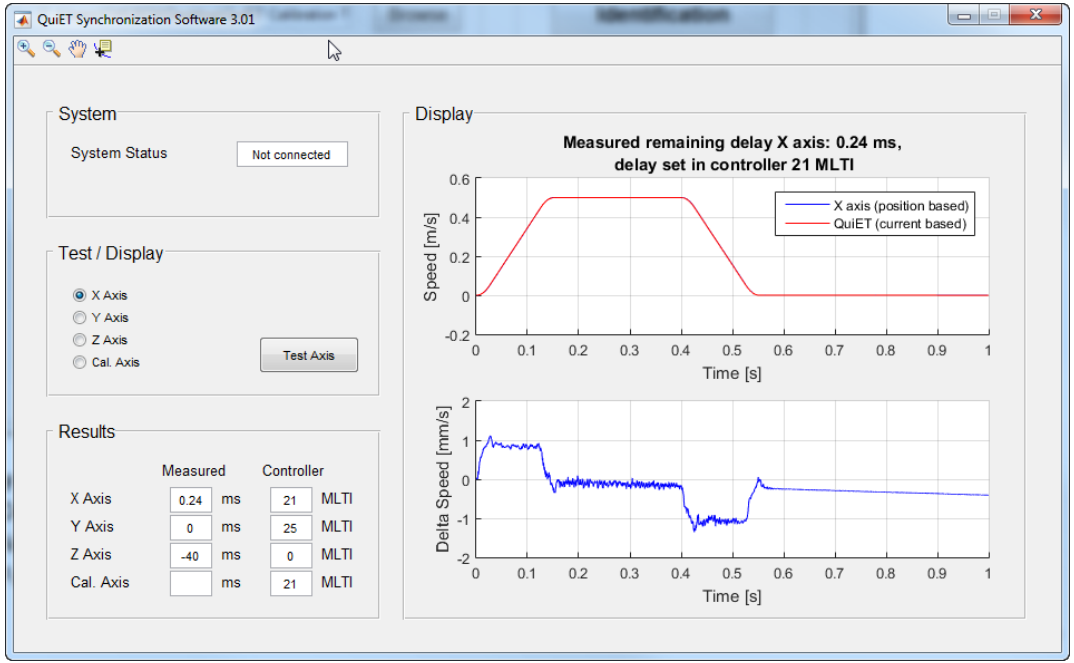


QuiET Commissioning Tool

Synchronization

Exact delay compensation required

- **Deterministic motion control architecture crucial for performance**



QuiET Commissioning Tool

Feed-Forward calibration

FF tuning example

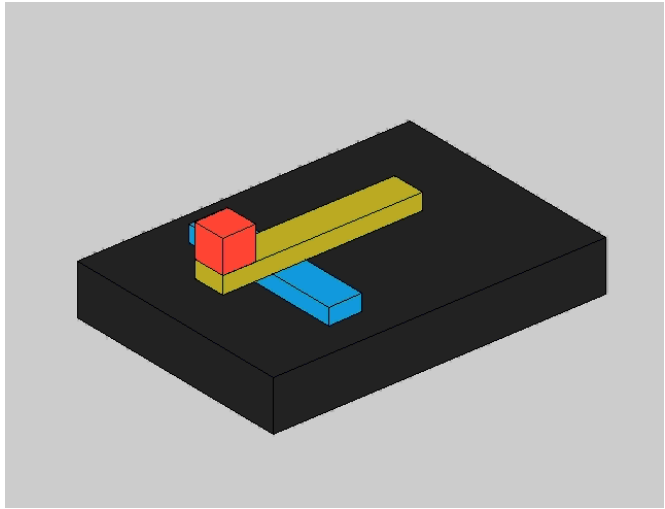
Tuning

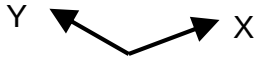
Model

- X, Y stage, X at the top
- Parameters to determine
  - Levers
  - Masses

Example

- M_x and $Xcog_z$
- Stage movement in X and monitor X and Ry of base



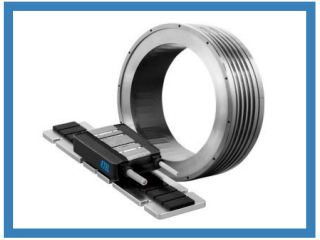


$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} m_{dx} \cdot \ddot{x} \\ \left(m_{dx} + m_{dy}\right) \cdot \ddot{y} \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} -\left(m_{dy} \cdot Ycog_z + m_{dx} \cdot Xcog_z\right) \cdot \ddot{y} \\ m_{dx} \cdot Xcog_z \cdot \ddot{x} \\ \left(m_{dy} \cdot Ycog_x + m_{dx} \cdot \left(x + Xcog_x\right)\right) \cdot \ddot{y} - m_{dx} \cdot \left(y + Xcog_y\right) \cdot \ddot{x} \end{bmatrix} + \begin{bmatrix} \left(m_{sx} + m_{sy}\right) \cdot y \cdot g \\ -m_{sx} \cdot x \cdot g \\ 0 \end{bmatrix}$$

Outline

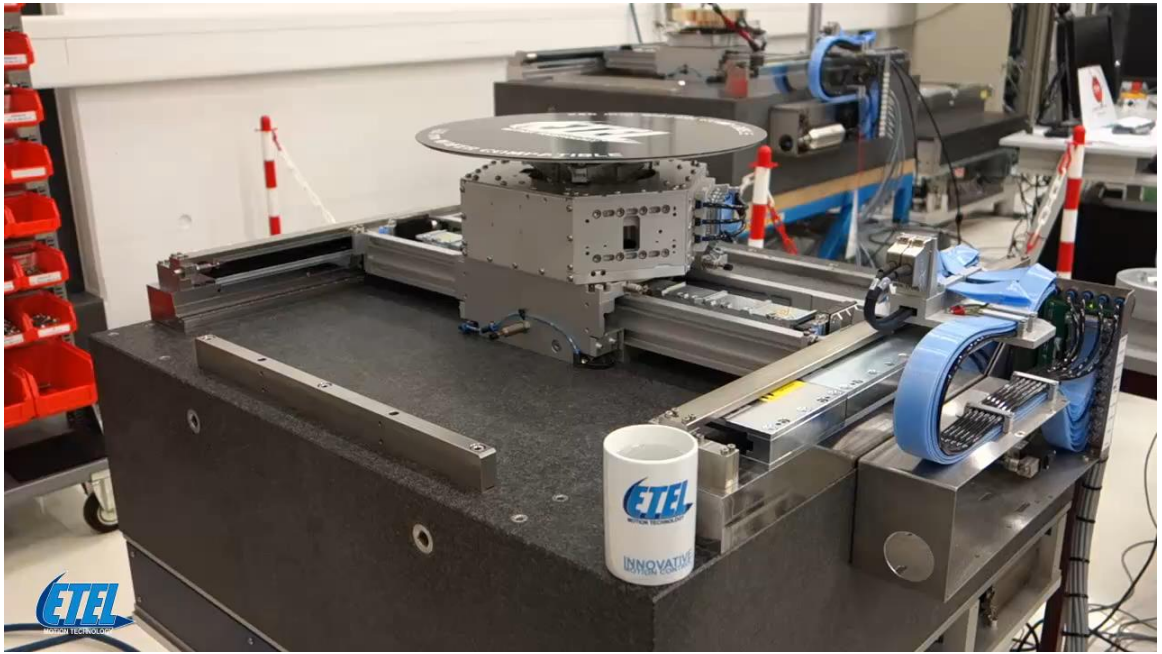
- Company presentation
- Product development workflow
- Active isolation systems
- Commissioning tool
- **Results**
- Conclusion



Performance : Qualitative

No QuiET (passive isolation only)

p = 0.04 m
v = 1 m/s
a = 1.2 m/s²



Performance : Qualitative

QuiET with 99% of force/torque cancellation

p = 0.2 m
v = 1 m/s
a = 6 m/s²

Conclusion (1/2)

ETEL gains

High-level programming in MATLAB and Simulink:

- Test many options before selecting a hardware & communication architecture
- Engineers focus on their core competence
- Use all powerful mathematical tools required for 6 DoF control
- Number of iterations between development teams reduced by around 80%

Common software interface with Simulink Real-Time and stand-alone products and deploying with MATLAB Compiler:

- Debug of commissioning algorithms during the development phase
- Avoid error-prone re-implementation

Conclusion (2/2)

Things to remember for the future

Use software modeling (e.g. UML) from the start:

- Use OOP to enable seamless interface swap
- Use a software architectural pattern (e.g. Model-View-ViewModel MVVM) to clearly separate the view (GUI) from the model

Automatic code generation for the controllers:

- Avoid error-prone re-implementation
- Avoid issues with quantization effects, etc.

Questions