

# T-iPPS: Semi-automated diagnostics

From weeks/months of analysis to hours of analysis

## Motivation

### Why diagnosis currently takes weeks to months:

- Missing insight in system execution
- Many (software) layers involved
  - Requires multidisciplinary and broad scope of system and domain knowledge → only a few people have this knowledge
- Many manual steps involved
  - Laborsome → high demand on a few people

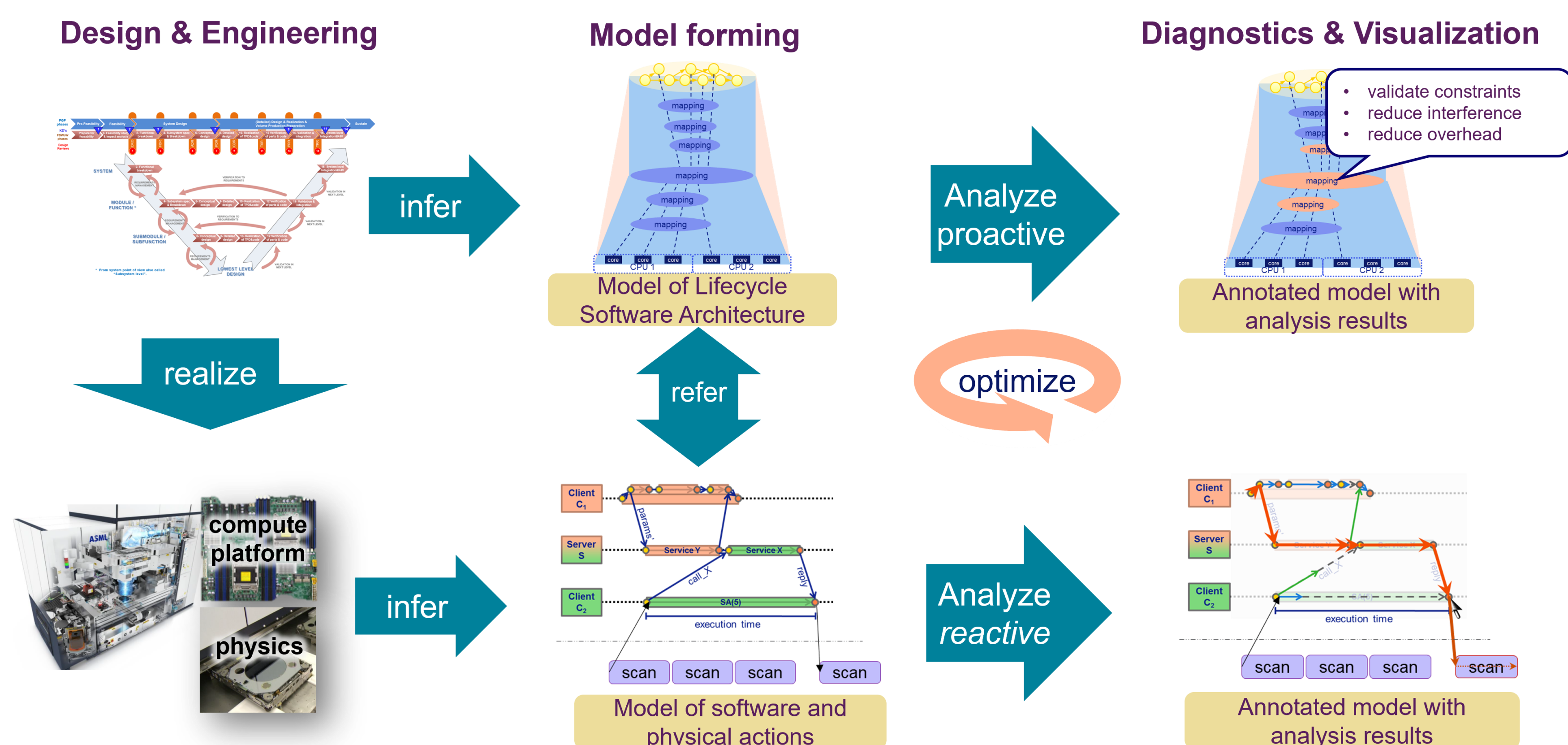
### Solution direction:

#### 1. Reduce Diagnosis time to hours, by tools that:

- Capture and tackle system complexity by incorporating multiple abstraction layers
- Capture knowledge in models and algorithms
- Semi-automate the analysis and identification of root-cause(s)

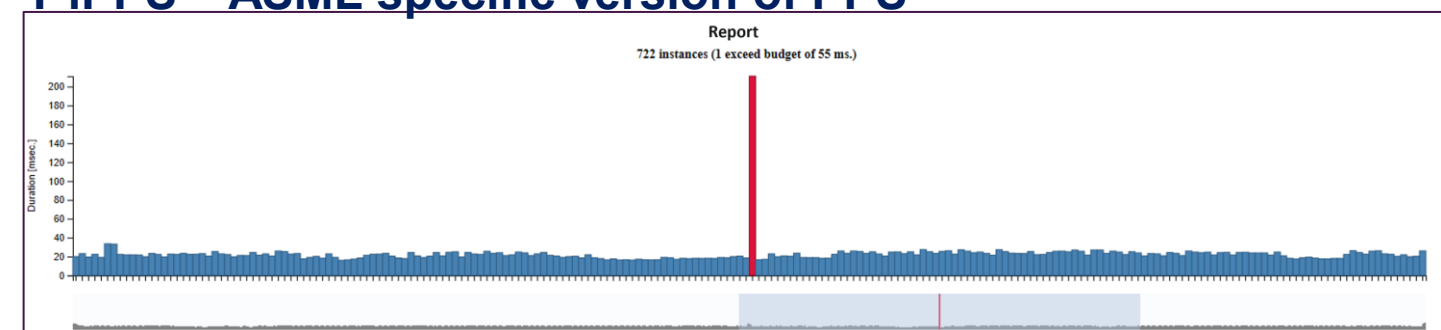
#### 2. From *Reactive* to *Proactive* diagnosis, enabled by the tools

## PPS method: Model-Based framework for semi-automated diagnostics

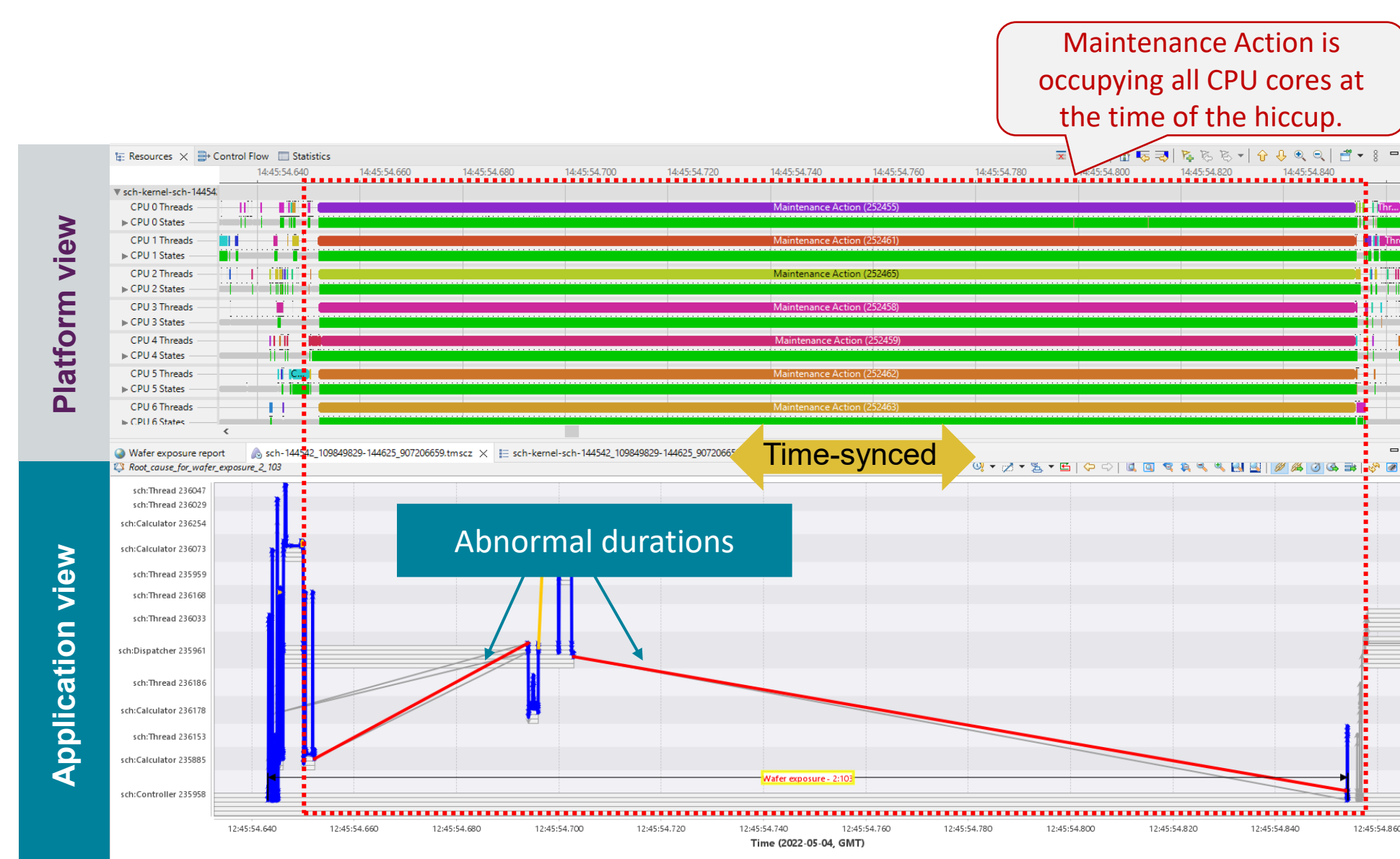


## Conclusion: Semi-automated diagnostics using PPS

### T-iPPS – ASML specific version of PPS



Here it is depicted, the **Automated Root-cause Analysis** using T-iPPS. T-iPPS is the ASML domain specific version of PPS, where the ASML tracing is translated to the ASML specific architecture concepts. This specific knowledge is captured as extension plugins on top of the PPS open-source toolset. The analysis techniques and frameworks are **generic** and non-domain specific, conforming to the **TMSC** formalism. On the left, after loading the execution tracing in the tool, a report is automatically generated showing the performance of all KPI instances in the form of bar chart. The bar of the instances that violate the budget are colored with red. When clicking on a red bar, the TMSC is loaded and the automated root-cause analysis algorithm pinpoints the dependencies of the TMSC with abnormal durations coloring them in red. Subsequently, the user can time-synchronize the TMSC (Application) view, with other system views, like the platform view and analyze why the durations of specific dependencies are abnormal.



### Useful links:

1. PPS tool: <https://esi.nl/research/output/tools/pps>
2. PPS tool download: <https://tno.github.io/PPS/>