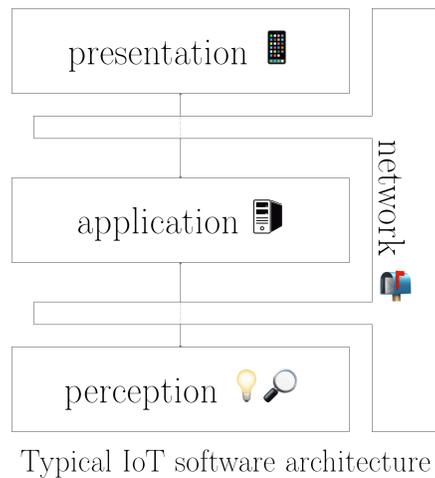
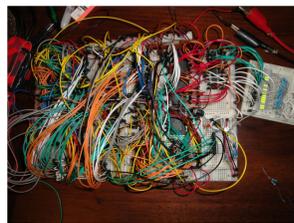


Challenges in the Internet of Things (IoT)



Typical IoT software architecture

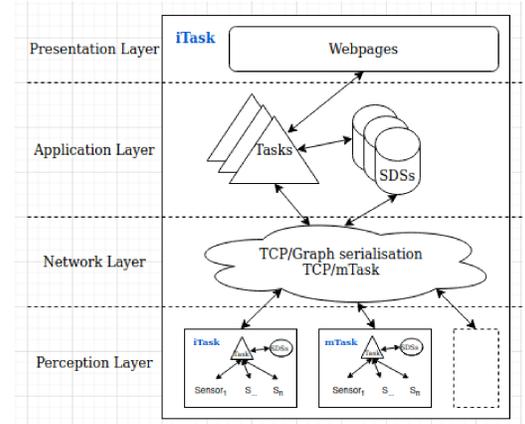
- ▶ Estimated 20 billion ($20 \cdot 10^9$) devices
- ▶ Heterogeneous architectures
- ▶ Semantic friction
- ▶ Interoperation problems



University of Glasgow case study

	tiered	tierless
single-board computer	PRS	CRS
micro controller	PWS	CWS

- ▶ real-world external example
- ▶ four-way comparison
- ▶ functional requirements
 - ▶ measure environment
 - ▶ centralised database
 - ▶ web interface to data
- ▶ management and monitoring of devices



Tierless (CWS, CRS) leads to 70%-90% reduction in SLOC compared to tiered (CRS, PRS).

Software metrics

Number of paradigms and languages.

Number of files and lines of code.

	tiered	tierless
category	PWS PRS	CWS CRS
edge	μ Python	Python
server	Python, JSON Redis, MongoDB PHP, HTML	iTask
total	7	6
para	imp. dec.	decl. decl.

	tiered	tierless
category	PWS PRS	CWS CRS
sensor node	52	11
sensor interface	178	9
communication	76	35
webinterface	56	28
database	106	12
management	94	5
total	562	100
files	35	3

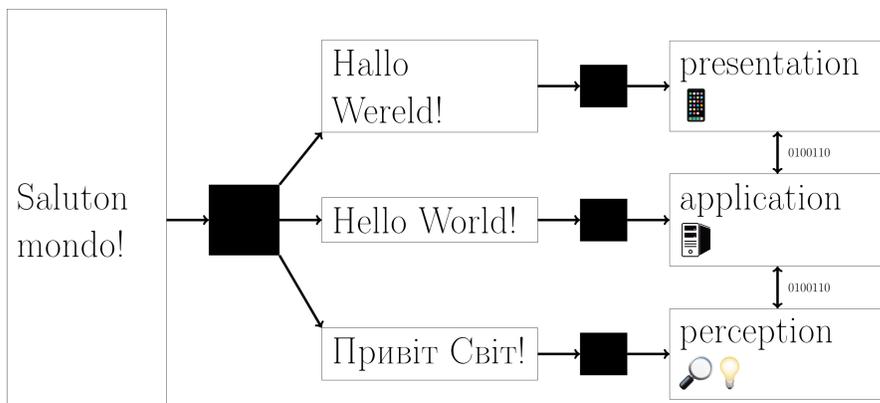
less paradigms/languages

less code

fewer interoperation problems

easier maintenance

Tierless programming



Typical tierless IoT architecture.

All layers and interoperation is generated from single: source, language, paradigm, and type system

Powered by task-oriented programming

The mTask language is a DSL implemented in Clean

Clean source code

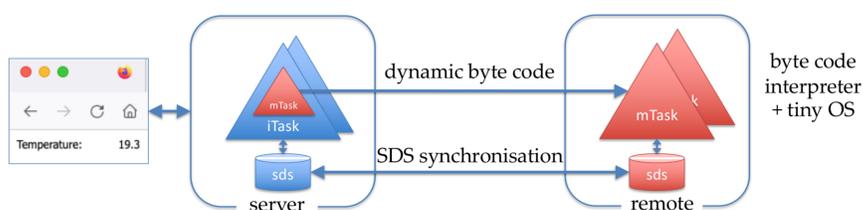
```
blink :: Int -> Main (MTask v Bool)
| mtask v
blink wait =
  declarePin D4 PMOutput \d4->
  fun \blinkfun=(\x->
    delay (lit wait)
    >>|. writeD d4 x
    >>|. blinkfun (Not x))
  In {main=blinkfun true}
```

mTask expression



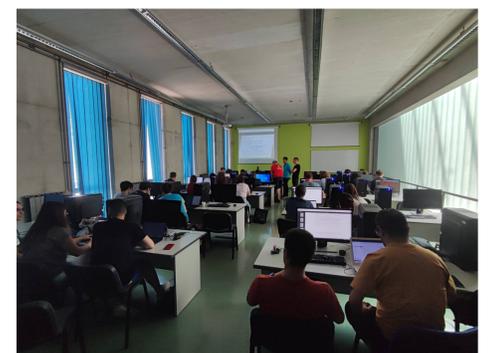
- Bytecode compiler
- Pretty printer
- Symbolic simulation
- Resource analysis
- ...

The mTask system integrates seamlessly with the iTask system.



Conclusions

- ▶ Tierless programming simplifies the creation and maintenance of IoT applications.
- ▶ Using DSLs, existing tierless languages can be extended to work on IoT edge devices.
- ▶ Proven itself in the UoG case study².
- ▶ Used by students in three summer schools: 3COWS 2019, SusTrainable 2022 and 2023.



SusTrainable 2023, Rijeka, Croatia

For more information



Reference 1



Reference 2



Get mTask here!

References

1. M. Lubbers, Orchestrating the Internet of Things with Task-Oriented Programming. in Radboud Dissertation Series, no. DIS-002. Nijmegen: Radboud University Press, 2023.
2. M. Lubbers, P. Koopman, A. Ramsingh, J. Singer, and P. Trinder, 'Could Tierless Languages Reduce IoT Development Grief?'. in ACM Trans. Internet Things, vol. 4, no. 1, Feb. 2023.

doi: 10.54195/9789493296114.

doi: 10.1145/3572901.