Cyber-Physical Systems Design: Foundations, Methods, and Integrated Tool Chains

John.Fitzgerald@ncl.ac.uk

Carl Gamble, Peter Gorm Larsen, Ken Pierce, Jim Woodcock

2008-2012: Industry deployment of advanced **engineering methods**

2010-2012: Collaborative modelling & co-simulation for **embedded systems**

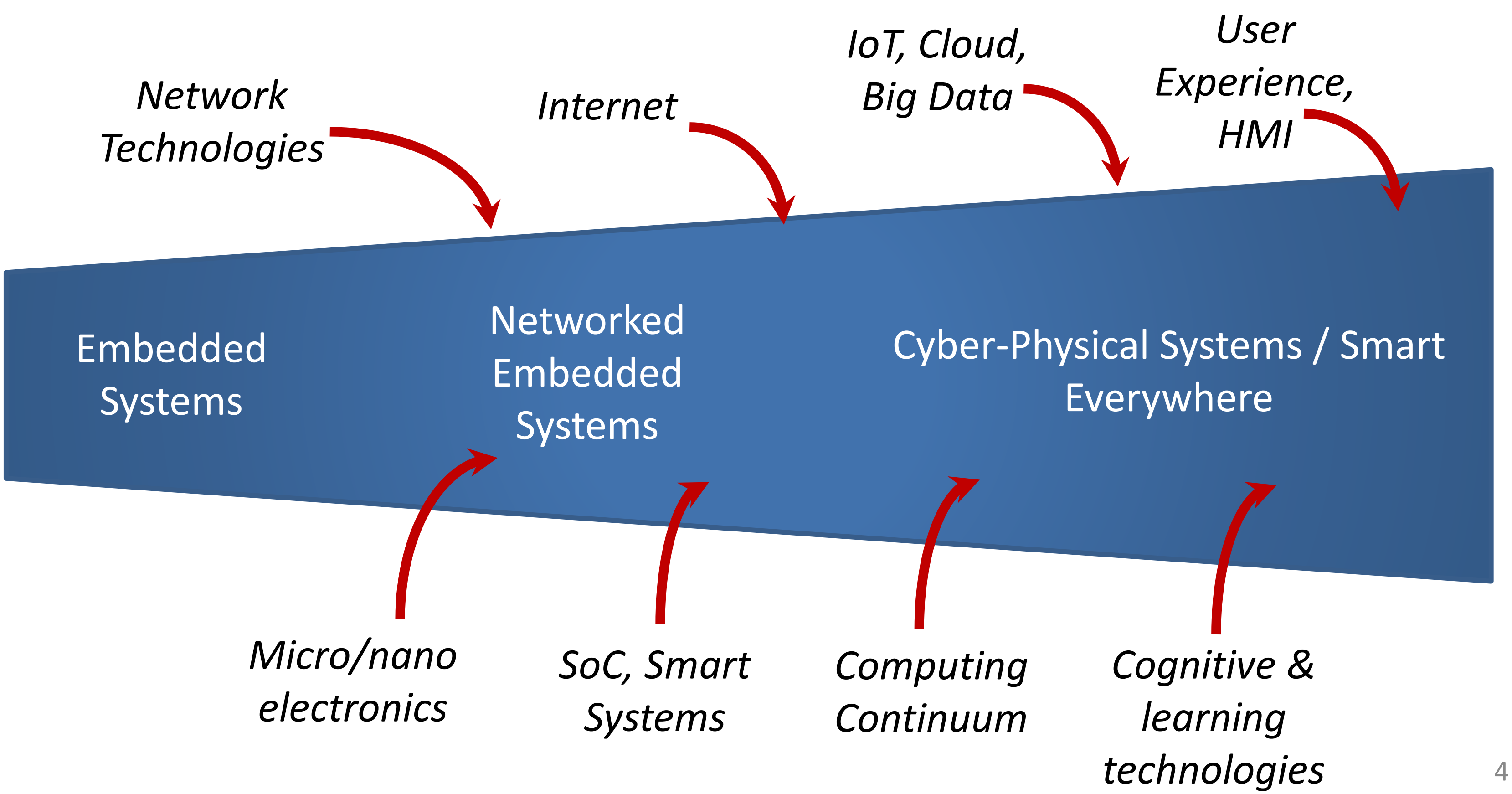
2011-2014: Methods & Tools for Model-based **Systems of Systems** Engineering

2015-2018+: **Cyber-Physical Systems** Engineering and Urban Systems.

1. Introduction
2. Basics
3. Three Key Features of a Solution:
   – Heterogeneous Modelling & Analysis
   – Exploring the CP Design Space
   – Traceability & Provenance in CPS Design
4. Concluding Remarks
5. Three short advertisements.

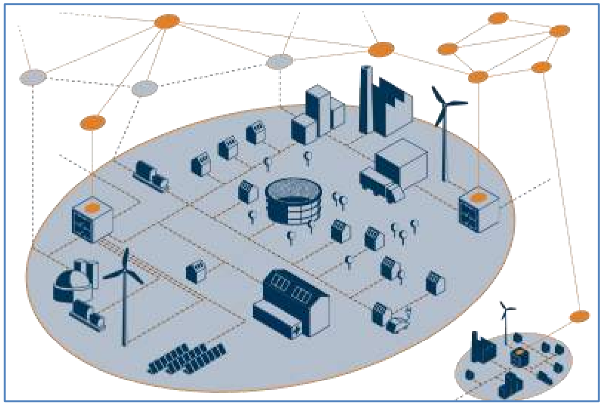
1. Introduction

Cyber-Physical Systems integrate computing and physical processes.



1. Introduction

Vehicle localisation
Obstacle detection
Brake assist
Fleet management
Congestion control
Toll payment



Emergency shutoff
Predictive maintenance
Fault detection
Virtual Power plant
Load prediction
Dynamic pricing

Technical Process
Organisational Process

Mastering the engineering and operation of high-performant CPS upon which people can depend

- **Integrated cross-domain architectures**
- Required **trustworthiness versus evolving** CPS
- **Design-operation continuum** (continuous deployment, live experiments)
- Engineering methods and tools able to **cope with the full scale and complexity of CPS**
- **Integrated cross-disciplinary models and analysis** for distributed analog/digital control and management
- **Human-technology interaction**

Source: CyPhERS project, 2014

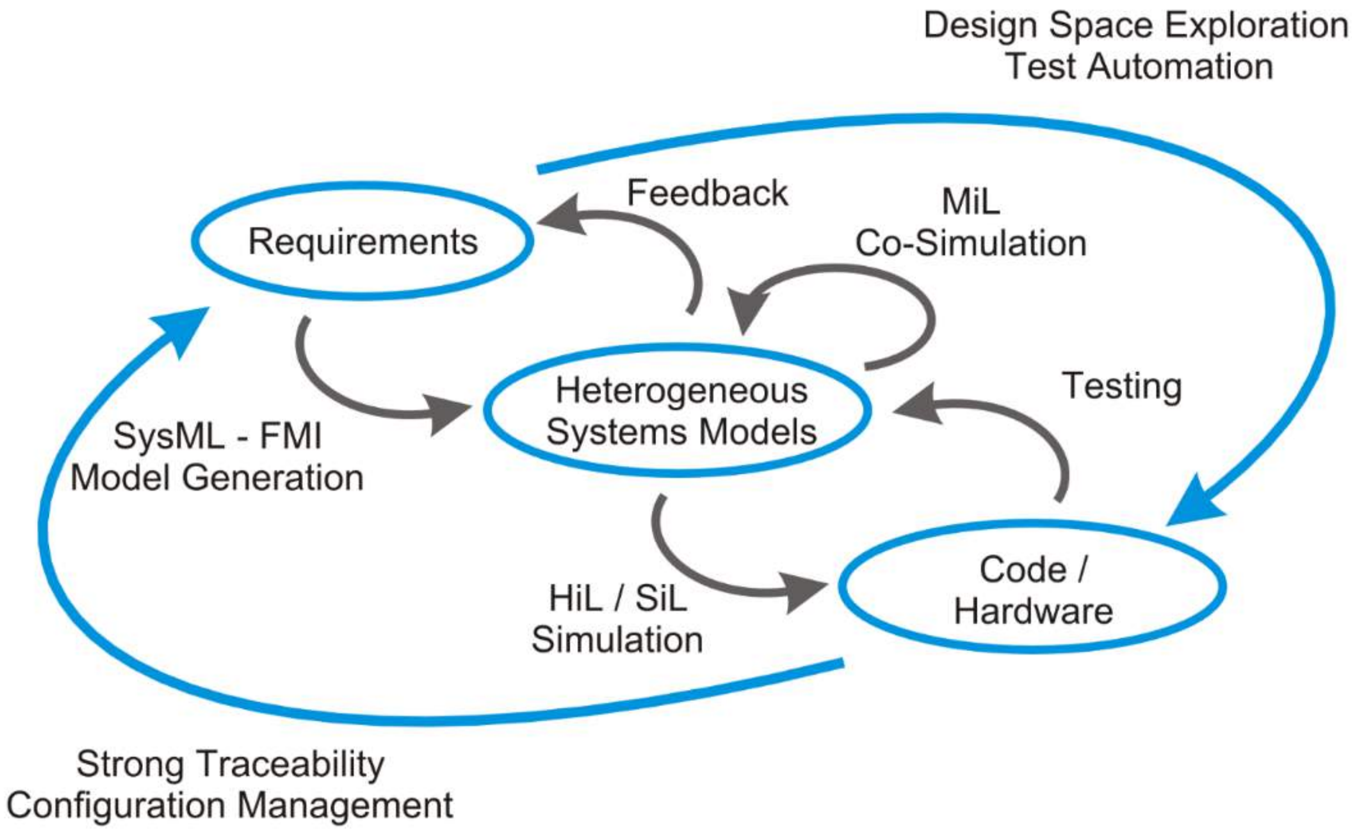
1. Introduction

Newcastle University

- CPS design necessarily multidisciplinary
  - Key properties are cyber-physical
  - Significance of supervisory control
  - Much software not written by software engineers!
- Key challenges:
  - **Foundations** addressing semantic heterogeneity
  - Model-based **Methods** for exploring design space
  - Not tools, but **Integrated Tool Chains**
- What would success look like?

1. Introduction

- Freedom for engineers to trade off across the cyber/physical divide, and to do so early.



2. Basics

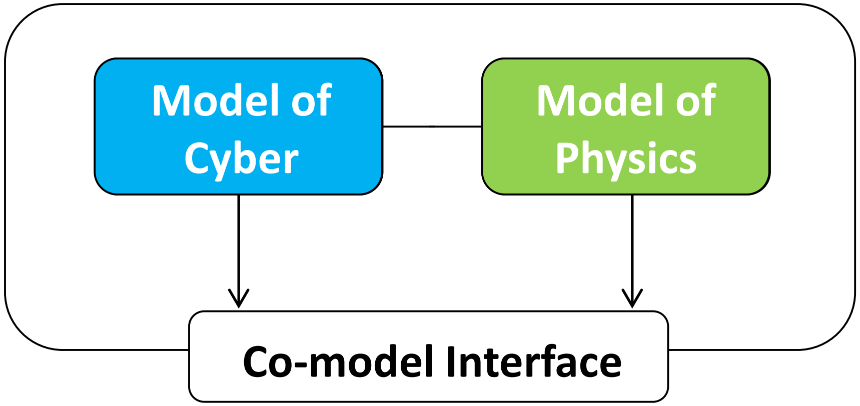
- System: collection of interacting elements organised for a stated purpose
- Dependable system: one on which reliance can justifiably be placed
- System of Systems (SoS):
  - Some elements are independently owned and managed systems, operating in their own right.
- Cyber-Physical Systems (CPS):
  - Some elements are computational processes and some are physical

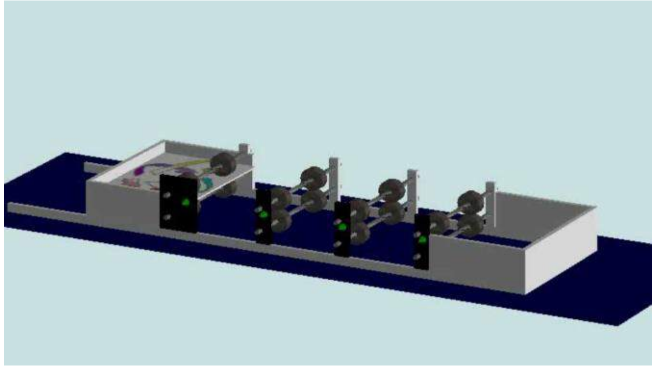
2. Basics

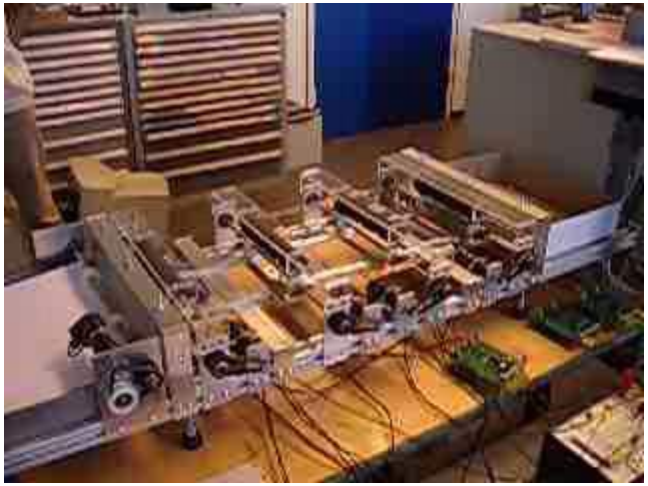
Software Models:
- Discrete
- Complex logic

Physics Models:
- Continuous
- Numerical

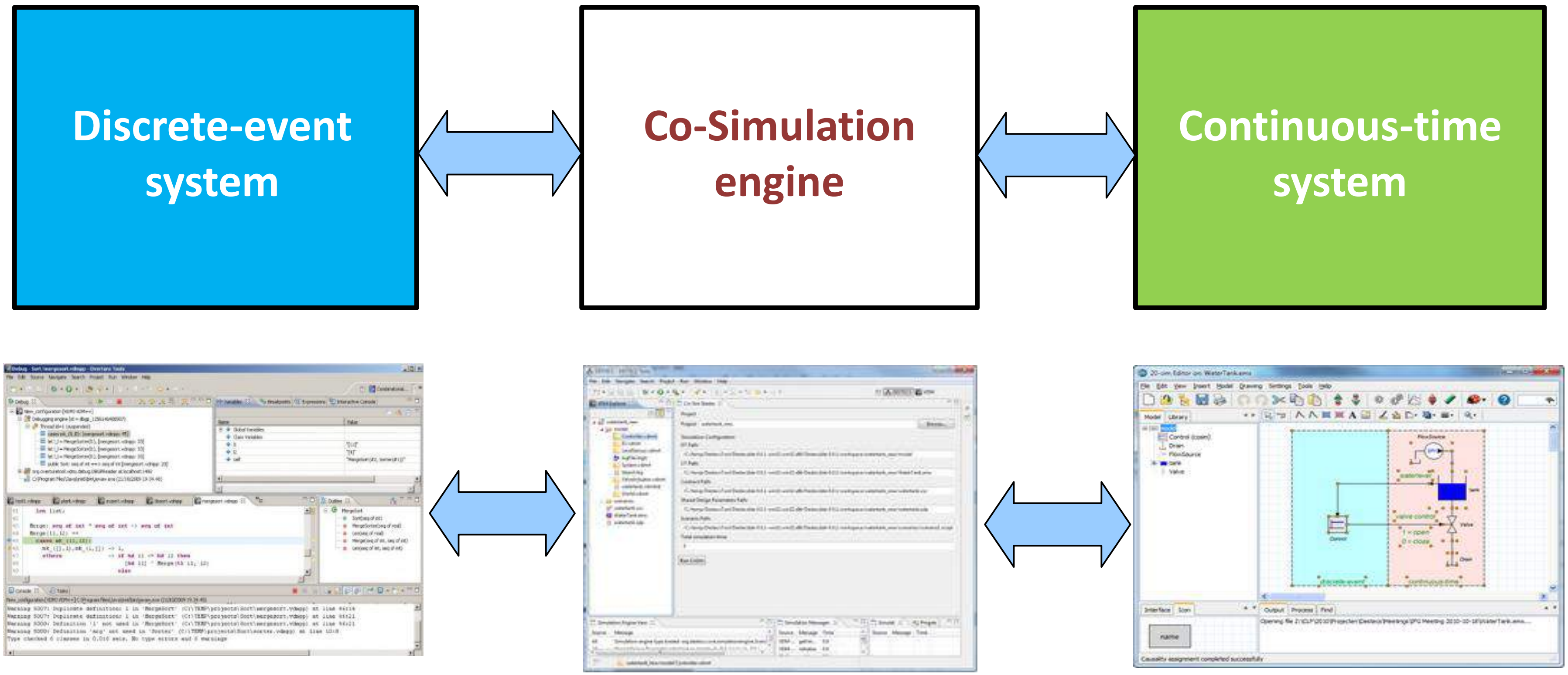
Co-model







2. Basics: co-simulation



Overture Crescendo 20-sim

2. CPS: co-model

DESTECS Project: Assisted mode for complex operations for a dredging excavator

Design Space Exploration optimised end-stop protection parameters

Koenraad Rambout (Verhaert): "A lot of time was saved on building physical prototypes. This ensures much faster iterations on physical models compared to traditional approaches. This enabled us to easily swap between different design solutions (e.g. hydraulic vs. electrical drives)"

2. Basics: co-modelling

- Tools (Crescendo) method guidelines (notably fault modelling); Automated Co-model Analysis (sweeps, ranking)
- Evidence that co-model-based design can work: Reduced design iteration/cost
- **But little networking, and design phases only**

1. Introduction
2. Basic Concepts
3. **Three Key Features of a Solution:**
   - **Heterogeneous Modelling & Analysis**
   - **Exploring the CP Design Space**
   - **Traceability & Provenance in CPS Design**
4. Concluding Remarks
5. Three short advertisements.

Heterogeneous Modelling

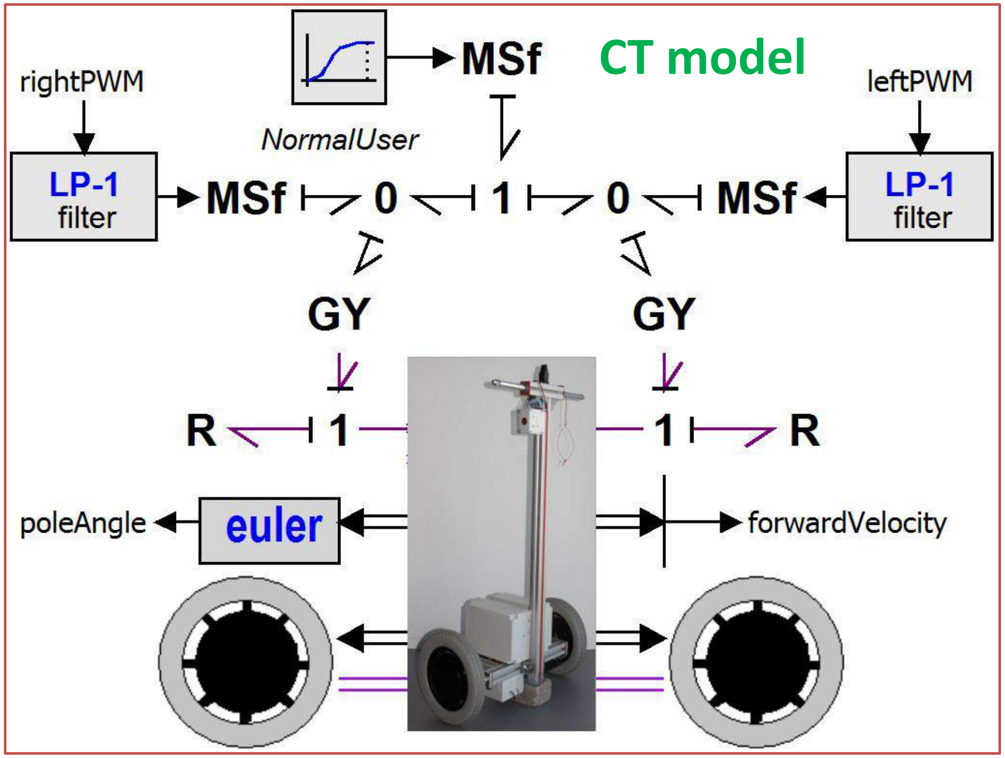
- Semantic Heterogeneity:
  - Across models
    - Discrete-event, continuous-time, stochastic, human, economic, …!
  - Between design tools
    - Co-simulator based on Structural Op. Sem.
    - Program Verifier based on axiomatic Hoare Calculus.
- No comprehensive formal foundations as yet.

Example: ChessWay

Demonstrated the need for co-modelling:

- High-fidelity physics model
- Low-level control loops OK, but need for DE abstractions (in VDM), e.g.
  - Modal behaviours
  - Fault Tolerance measures
- Not always clear where to model (e.g. human behaviours)

Example: ChessWay



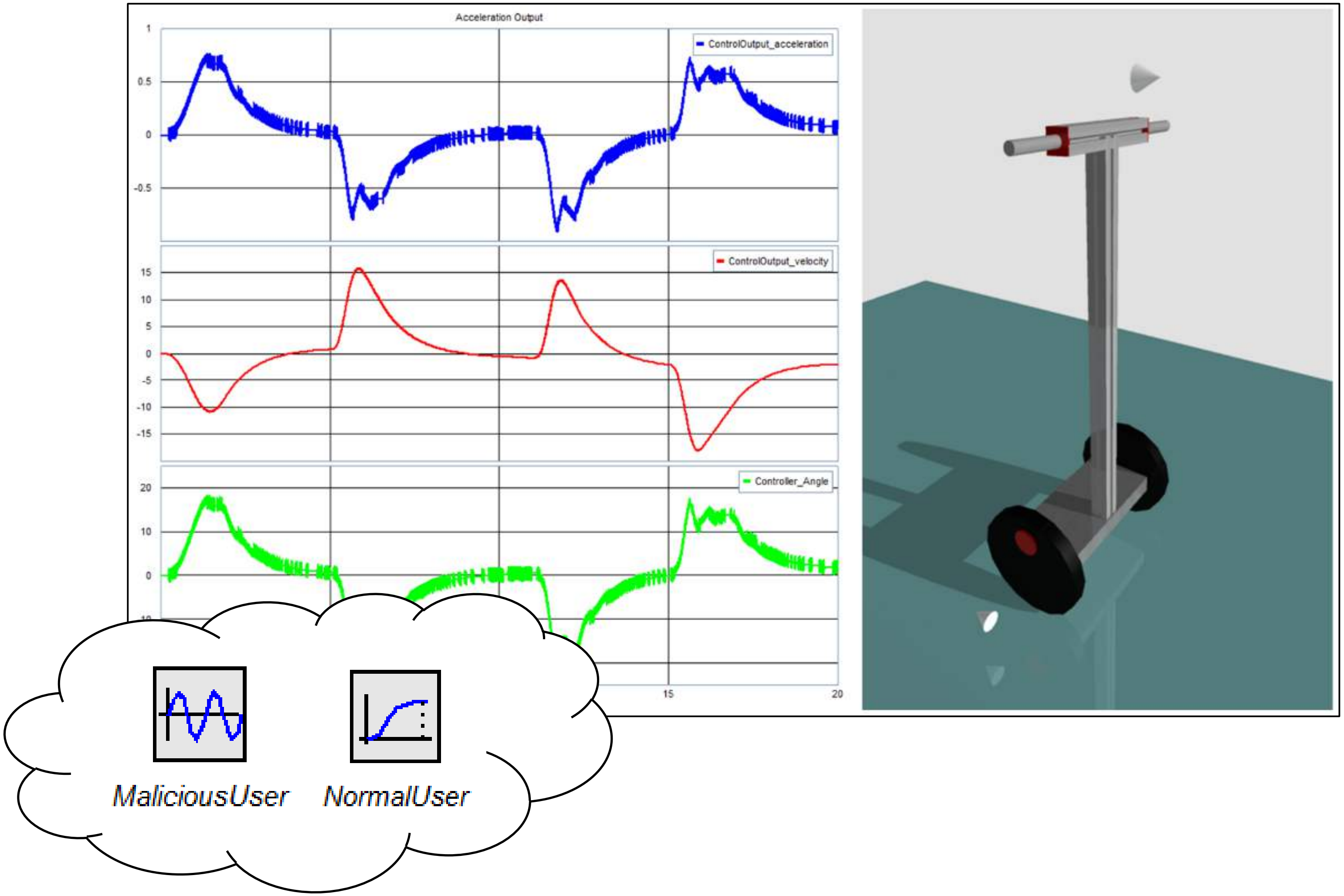
	Name	Type	Notes
controlled	leftPWM	real	range: [-1,1]
	rightPWM	real	range: [-1,1]
monitored	poleAngle	real	range: $[0,2\pi]$
	forwardVelocity	real	

DE model

```
class Controller

instance variables
  -- sensors
  private angle: real;
  private velocity: real;
  -- actuators
  private acc_out: real;
  private vel_out: real;
  -- PID controllers
  private pid1: PID;
  private pid2: PID;

operations
  public Step : () ==> ()
  Step() == duration(20) (
    dcl err: real := velocity - angle;
    vel_out.Write(pid2.Out(err));
    acc_out.Write(pid1.Out(angle));
  );

  public GoSafe : () ==> ()
  GoSafe() == (
    vel_out.Write(0);
    acc_out.Write(0);
  );

thread
  periodic(1E6,0,0,0)(Step); -- 1kHz

end Controller
```

Interface “Contract”

Exploring the Design Space



Example: ChessWay

DESTECS Project: The ChessWay Personal Transporter
Early detection of design errors

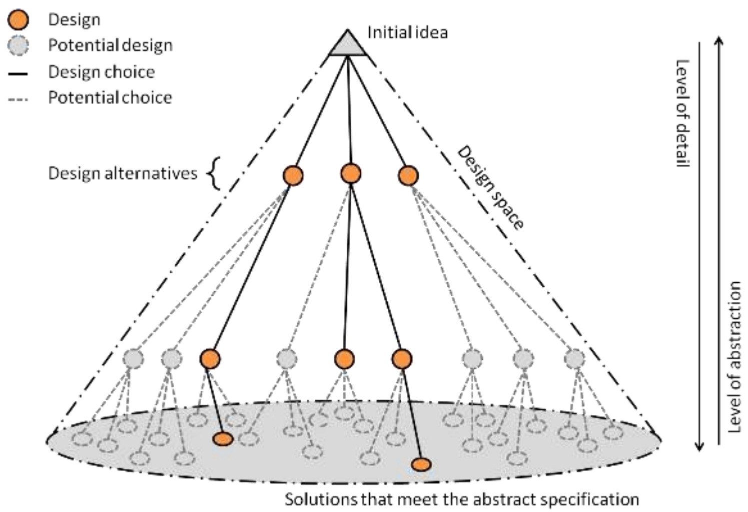
Bert Bos (Chess): "Debugging in the co-simulation environment is much quicker than debugging real-time embedded control software. … the initial implementation worked the first time… fault handling usually takes several cycles to work properly."

Semantic Heterogeneity

New Challenges

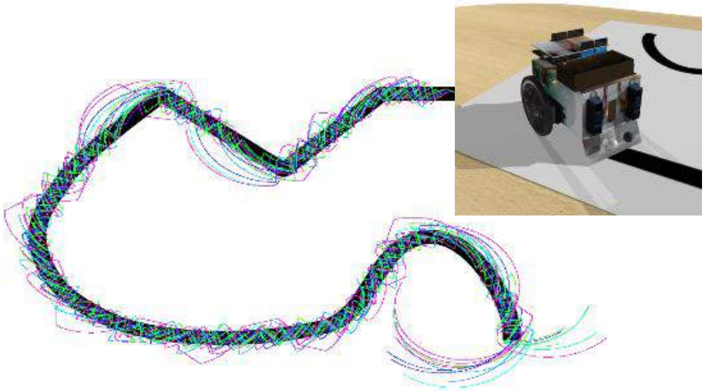
- Currently exploring Unifying Theories of Programming
  - Computation Paradigms: Object-oriented, concurrent, real-time, discrete, continuous, …
  - Abstraction
  - Presentation (Operational, Algebraic, Axiomatic, Denotational)
- Some success in COMPASS for SoS.

Exploring the Design Space



- Systematic exploration of solution space
- Optimisation against defined criteria
- Ranges of design parameters
- Ranking of design alternatives
- Or further genetic or evolutionary optimisation on a Pareto front.

		Metric*				
Rank	Design	A	B	C	D	Mean Rank
1	**(b)**	**1**	**5**	**1**	**2**	**2.2**
2	(f)	7	2	4	1	3.5
3	(a)	2	8	2	4	4.0
4	(e)	3	6	3	5	4.2
5	(i)	9	1	5	3	4.5
6	(c)	5	3	6	8	5.5
7	(d)	6	4	7	7	6.0
8	(h)	4	7	8	9	7.0
9	(j)	8	9	9	6	8.0

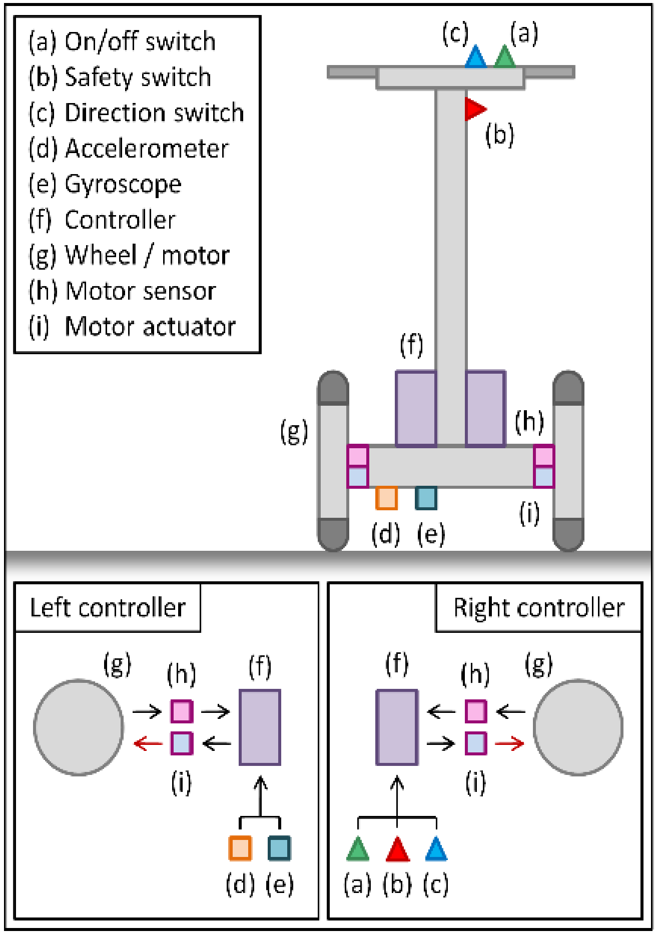


A = distance, B = energy,
C = deviation area, D = max. deviation

Exploring the Design Space

Example: a wireless ChessWay?

- What control loop frequencies provide safe balancing?
- Consider alternative frequencies and allocations of responsibilities between controllers.
- Determine how lossy comms can be maintaining safety conditions.
- You can have a wireless ChessWay if loss <15% ☺



Exploring the Design Space

New Challenges:

- For DSE, performance is critical.
- Tacit knowledge and gut instinct are important to narrow the space – can we augment these with reasoning, e.g. from test automation?

Traceability & Provenance

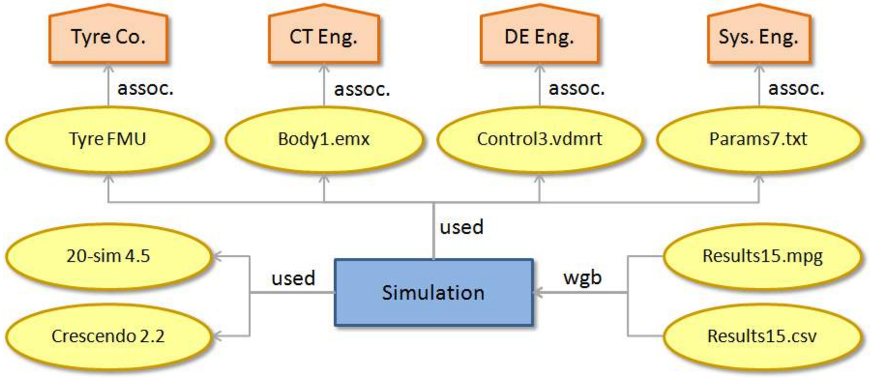
- In a co-model-based development, very diverse forms of evidence are produced.
- Marshalling complexity
- Ramifications of change
- Traceability documentation often dropped under pressure!

Traceability & Provenance

- Standardised provenance structures can show dependencies in design set
- Consider a change of tyre supplier for the ChessWay (e.g. compiled FMU)

New Challenges:

- Richer design sets
- Provenance graphs for co-modelling in Prov-N
- Graph abtractions for managing complexity

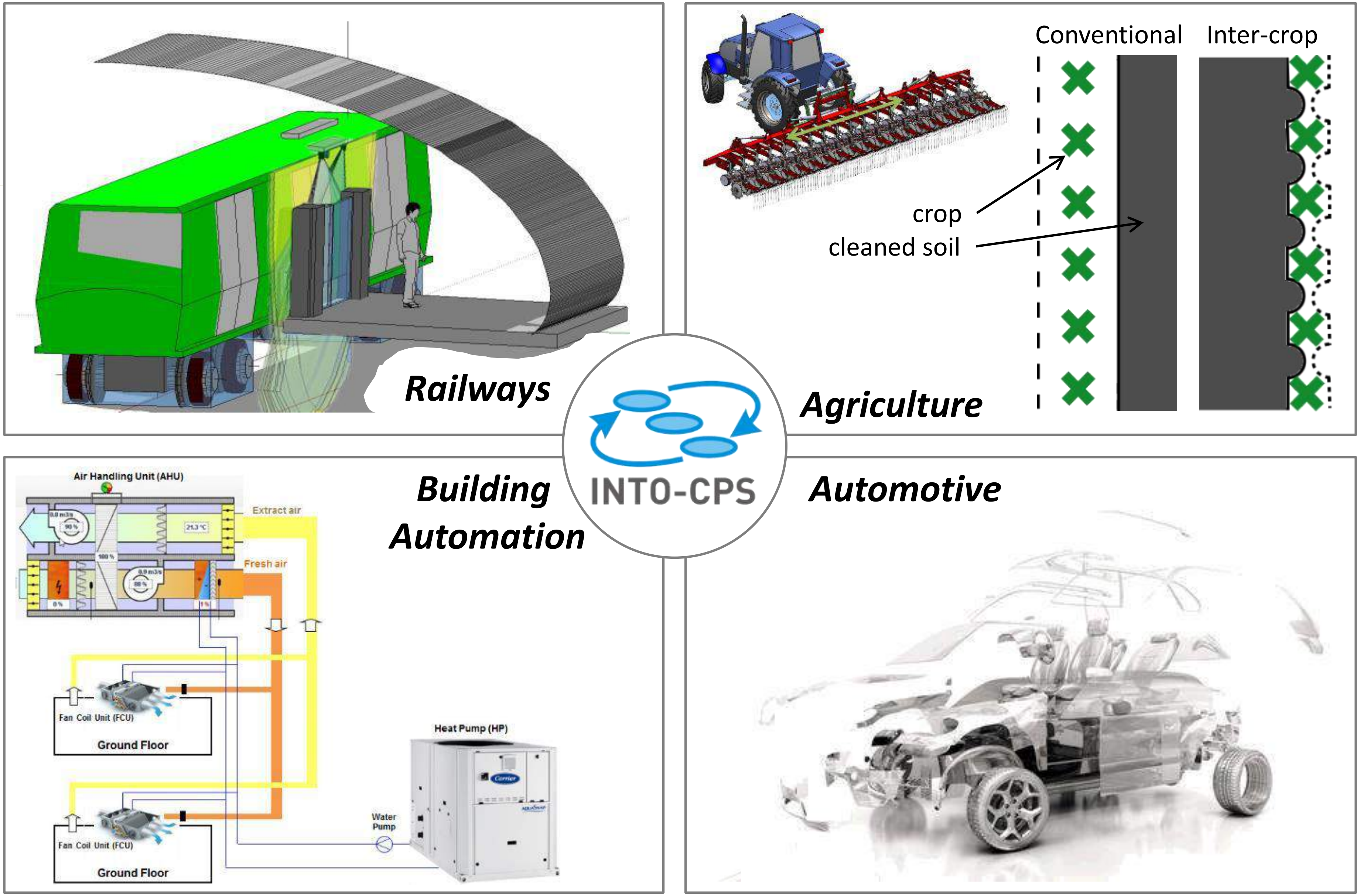


Concluding Remarks

- **Multidisciplinary** model-based design of CPSs is inherently collaborative
- It’s **not only about dependability**, but about reducing development risk and time to market
- **Formal foundations** are needed to address semantic heterogeneity within co-models and across tools.
- **Formal techniques** have much to offer exploration of the design space
- **Formal approaches** to managing evidence in the design set are needed to help construct sound tool chains.

Short Advert 1: INTO-CPS

- http://into-cps.au.dk
- Well-founded tool chains, not a single factotum tool:
  - Foundations in UTP
  - Static analysis of co-models
  - Requirements, Architectures (SysML) to code
- Baseline Technologies:
  - Modelio, VDM, 20-sim, Open Modelica, TWT co-sim engine, RT Tester.


Railways
Conventional
Inter-crop
crop
cleaned soil
Agriculture
INTO-CPS
Building
Automation
Automotive
Air Handling Unit (AHU)
Extract air
Fresh air
Fan Coil Unit (FCU)
Ground Floor
Fan Coil Unit (FCU)
Ground Floor
Heat Pump (HP)
Water
Pump

Advert 2: CPSE Labs

- www.cpse-labs.eu
- H2020-ICT-2014-1 – Innovation Action, 36 months
- Eight core partners in five countries
- Expediting and accelerating the realisation of trustworthy CPS
  - Foster pan-European **network of design centres** committed to transitioning science and technology for … dependable CPS
  - Identify, define, and execute focused and fast-tracked **experiments**
  - Spread best CPS engineering practices and **learning among industry and academia**
  - Establish a **marketplace** for CPS engineering assets

CPSE Labs Design Centres

Competencies and Application Domains

Centre UK

Newcastle University

CPSE Competencies:
- Model-based engineering
- Co-modelling & simulation
- Industrial formal techniques

Application Domains:
- Urban CPS
- Environment & Sustainability

Centre Sweden

KTH

CPSE Competencies:
- Model-based engineering
- Integrated engineering environments
- Autonomous machines

Application Domains:
- Automotive
- Production systems

Centre Germany North

OFFIS

CPSE Competencies:
- HW/SW co-design
- E/E architectures
- Model-based safety & security analysis

Application Domains:
- Maritime

Centre Spain

indra

CPSE Competencies:
- Internet of Things
- Geospatial technologies
- Transportation Systems
- Cloud Services

Application Domain:
- Smart City

Centre France

ONERA THE FRENCH AEROSPACE LAB LAAS-CNRS

CPSE Competencies:
- Robotic SW architectures
- Safety assessments

Application Domains:
- Aerospace
- Robotics
- Automotive

Centre Germany South

fortiss

CPSE Competencies:
- Model-based engineering
- Flexible production systems
- Internet of Things

Application Domains:
- Automotive
- Production systems
- Avionic

CPSE Labs

Experiments

First Call
open **now!**

- Projects with a specific *innovation objective*
  - Fast-track (12-18 month) and focused (3-6 partners)
- Three rounds of *open calls*
  - At ~M3, M9, M18
- Cost €150k max. per third party
- Centres have PMs to help in experiments

What is the Point?

Newcastle Science Central: Digitally Enabled Sustainability

- £58m investment: building now
- **Core programme: Digitally Enabled Urban Sustainability**
- Urban Sustainability problems require collaborative systems solutions:
  – Technical Interventions
  – Community decision making (Digital Civics)

1970s (Reactive)


Newcastle 'greenest' British city

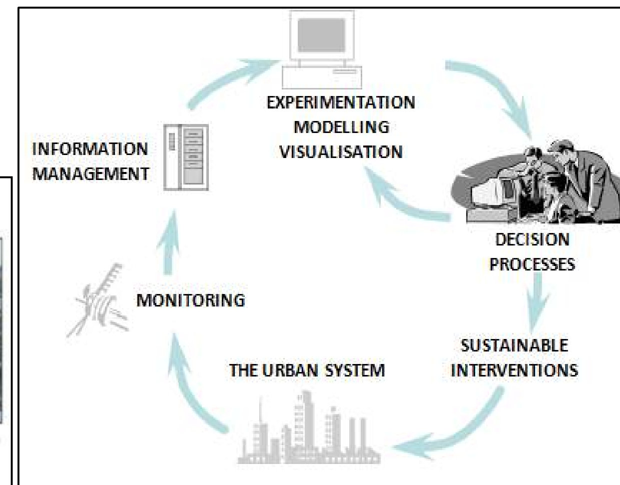
Newcastle upon Tyne has been named as Britain's greenest city in a think tank's annual study.

Forum for the Future looked at the sustainability of the 20 biggest cities, measuring factors such as air quality, wildlife and quality of life.

As well as greenest city, Newcastle was the overall most sustainable, beating 2008 winner Bristol into second.

Newcastle was praised for emerging from its industrial past to go green

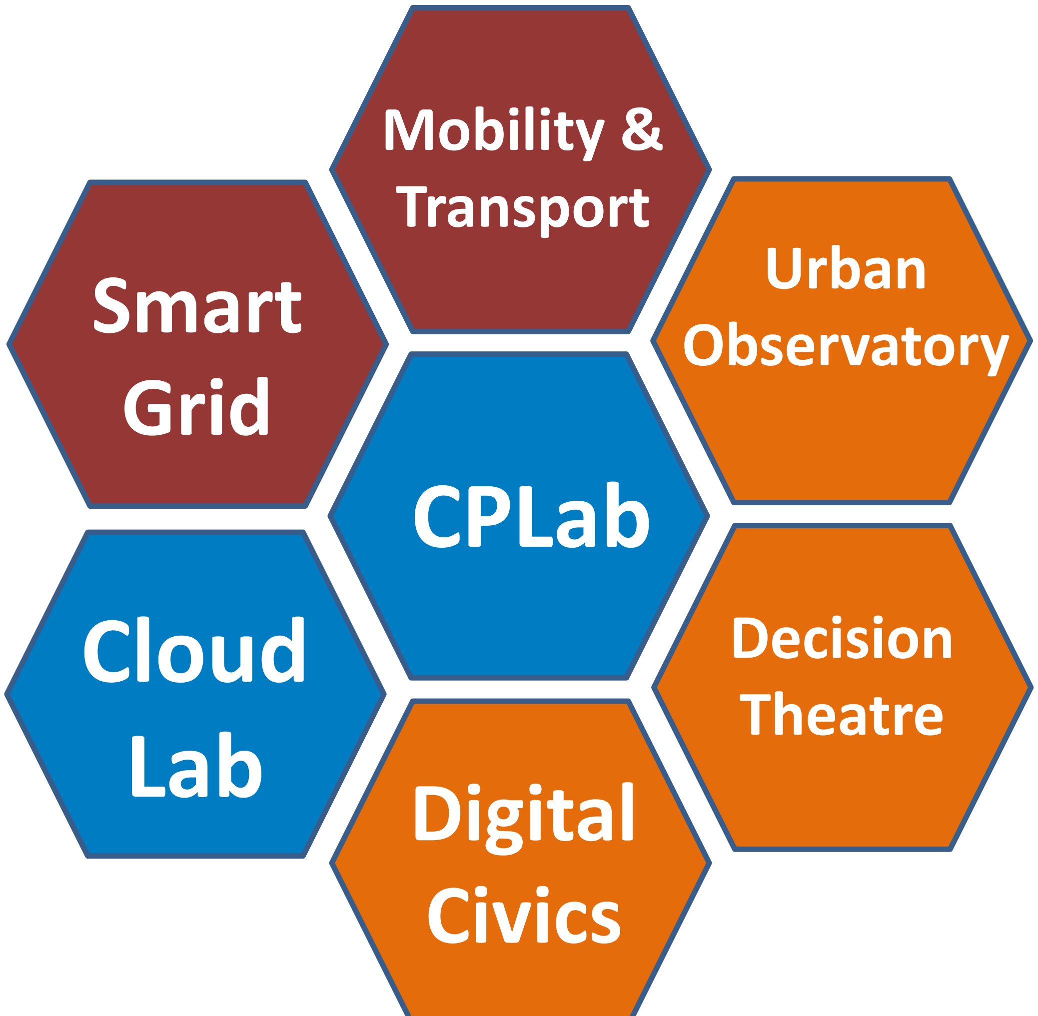
2010 (intelligence)



2050

CPLab Newcastle

Newcastle University



Newcastle University

Northern Powergrid

Siemens

Philips

Northumbrian Water living water