Holistic, Distributed Stream Processing in IoT Environments

Peter Michalák
CDT Cloud Computing for Big Data

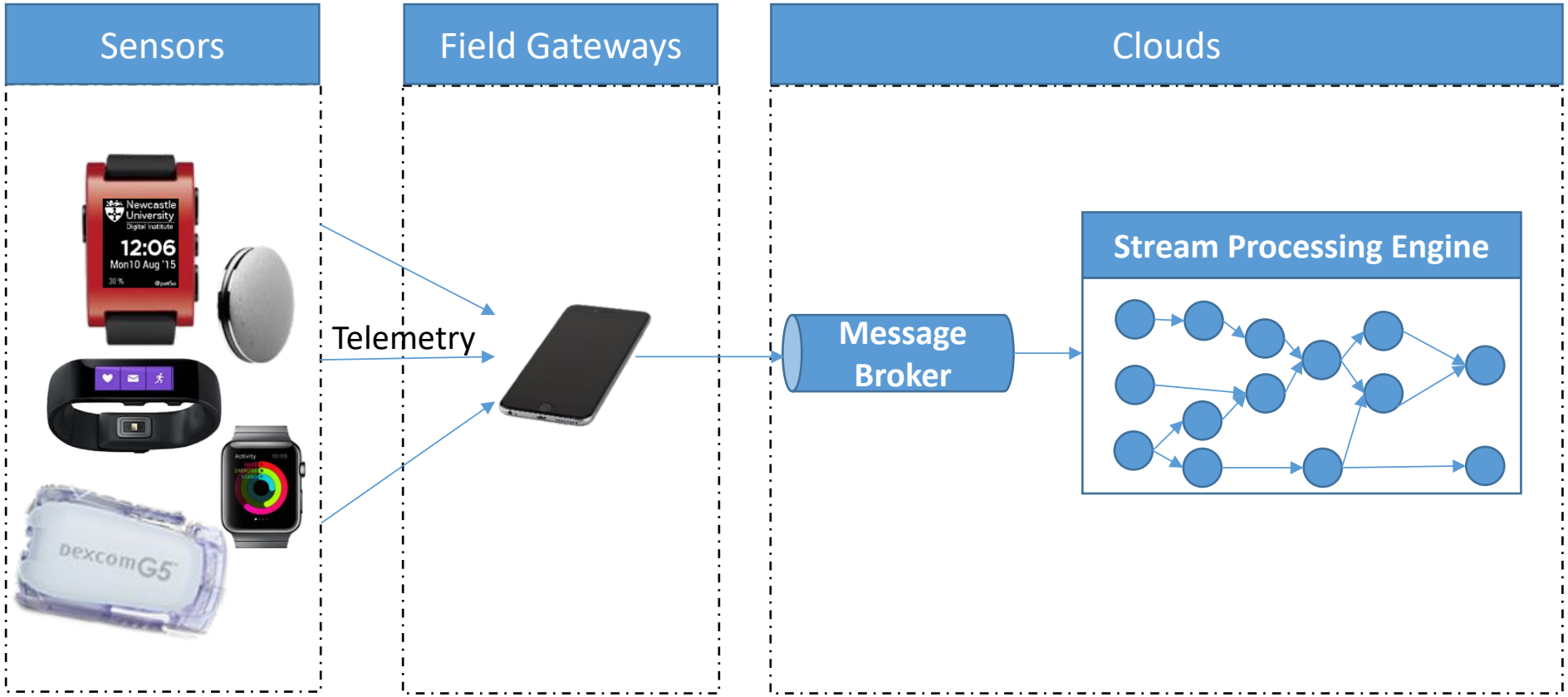
Prof. Paul Watson
School of Computing

Dr. Sarah Heaps
School of Maths and Stats

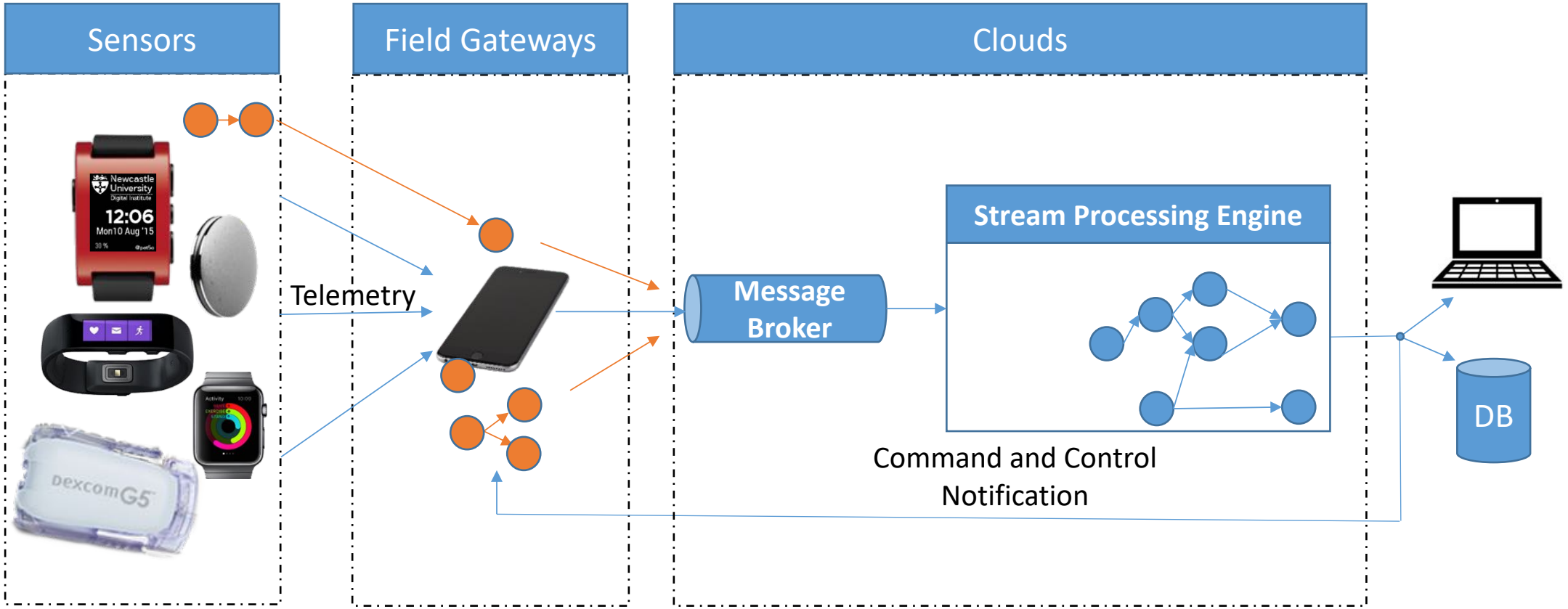
Prof. Mike Trenell
Medical School

23. 03. ‘18 UK Systems Research Workshop

Stream Processing in IoT

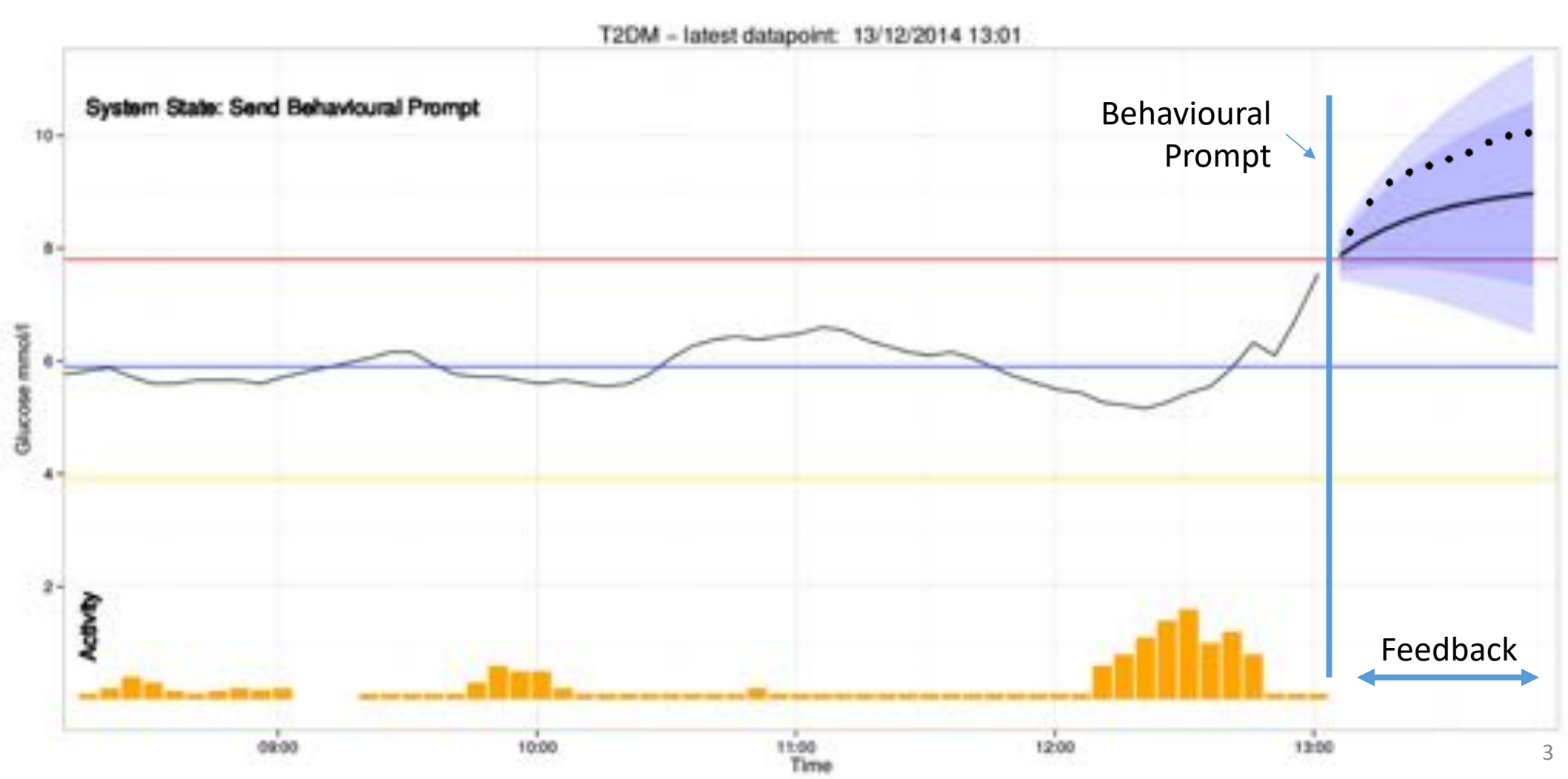


Holistic Stream Processing in IoT



Healthcare use case

Behavioural Prompts & Feedback



Challenges

- Energy
  - Pebble Watch battery life ~7 days
  - Streaming raw accelerometer data reduces battery life ~18 hours

ESPer/Spark/Storm
sky is the limit

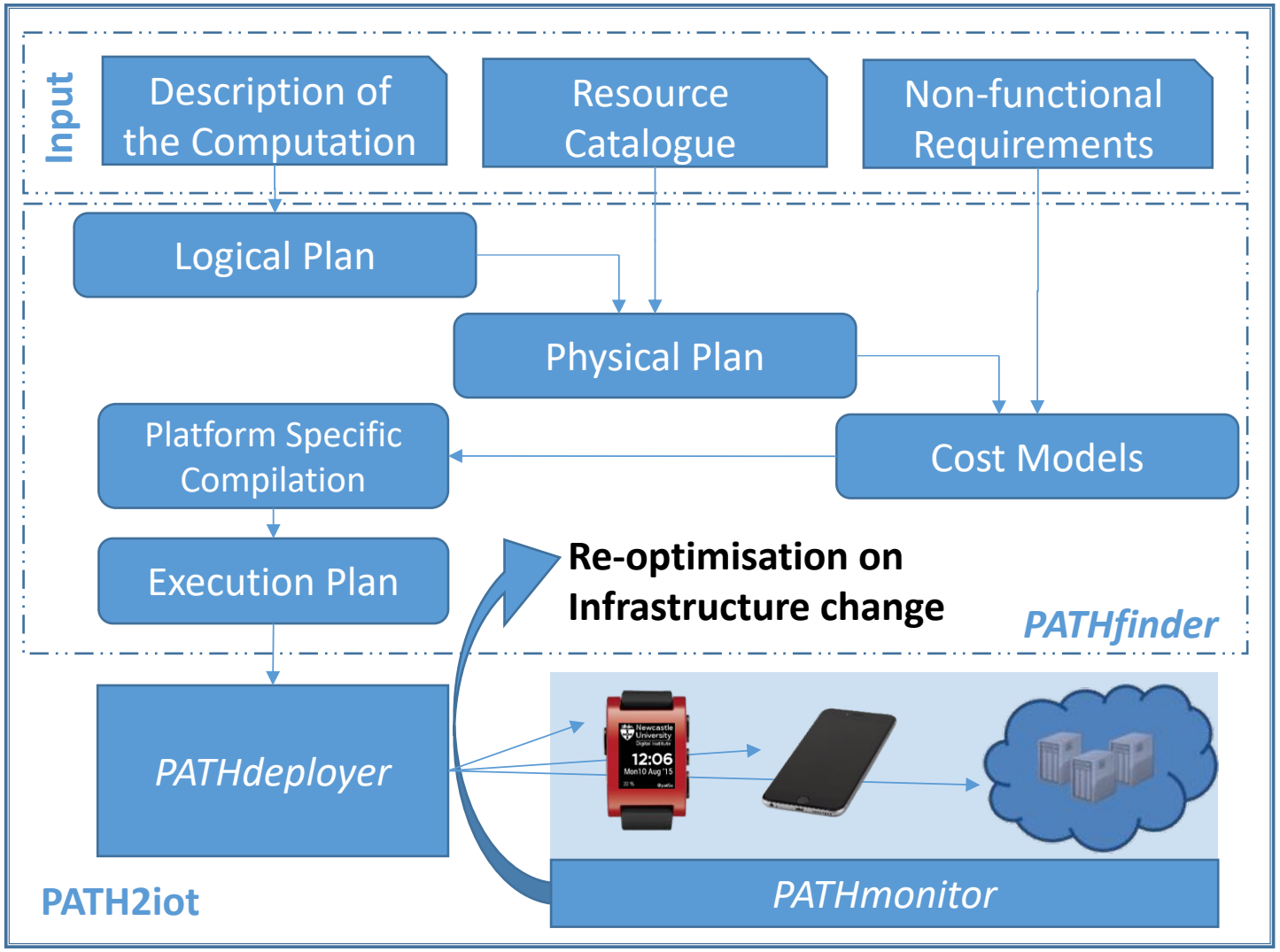
- Hand-crafting bespoke solutions
  - Definition of data stream processing

Objective C / Swift
Java / Kotlin ..

- Programming for Heterogeneous Platforms
  - Multitude of devices, APIs and programming languages

C / JavaScript

PATH2iot system
Automating Computational Placement



- *PATHfinder*
  - Automated Computational Decisions
  - Non-functional requirements
  - Device-specific compilation
- *PATHdeployer*
  - A deployment tool delivers configuration to enable computation
- *PATHmonitor*
  - Future work for PATH2iot

PATHfinder : High-Level Declarative Description of Computation

Step count algorithm[1] in EPL

1. INSERT INTO AccelEvent
   SELECT getAccelData(25, 60) FROM AccelEventSource
2. INSERT INTO EdEvent
   SELECT Math.pow(x*x+y*y+z*z, 0.5) AS ed, ts
   FROM AccelEvent WHERE vibe=0
3. INSERT INTO StepEvent
   SELECT ed1('ts') as ts FROM *EdEvent*
   MATCH RECOGNIZE (MEASURES A AS ed1, B AS ed2 PATTERN (A B)
   DEFINE A AS (A.ed > THR), B AS (B.ed ≤ THR))
4. INSERT INTO StepCount SELECT count(*) as steps FROM
   StepEvent.win:time_batch(120 sec)
5. SELECT persistResult(steps, "time_series", "step_sum") FROM StepCount

- Event Processing Language (EPL) from Esper
  - High Level Declarative Description of Computation
  - SQL based with extended grammar to support CEP operations
  - Decomposable into directed graph of stream operators

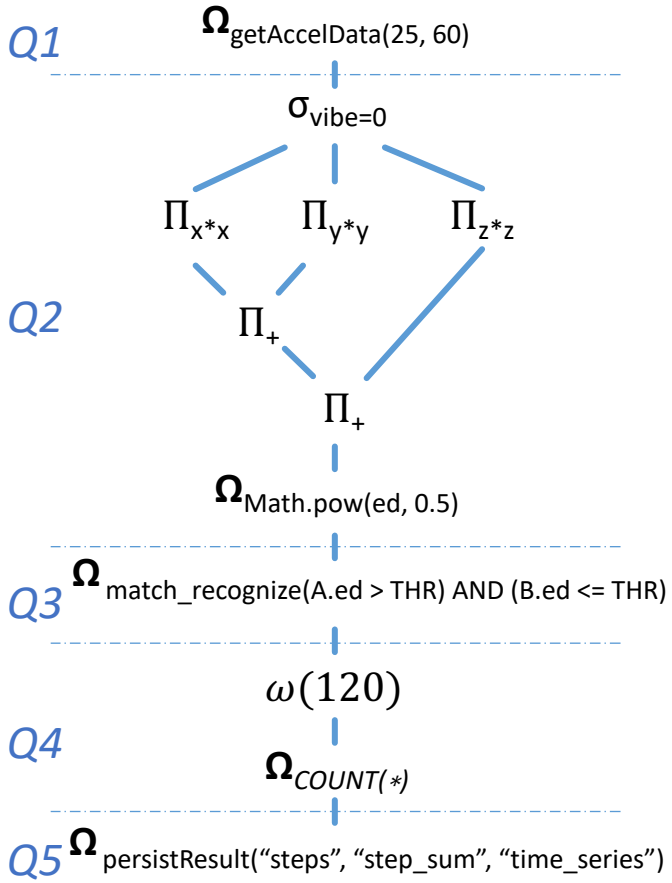
[1] N. Zhao, "Full-featured pedometer design realized with 3-axis digital accelerometer," *Analog Dialogue*, vol. 44, no. 06, 2010.

PATHfinder : High-Level Declarative Description of Computation

Input: Description of the Computation | Resource Catalogue | Non-functional Requirements

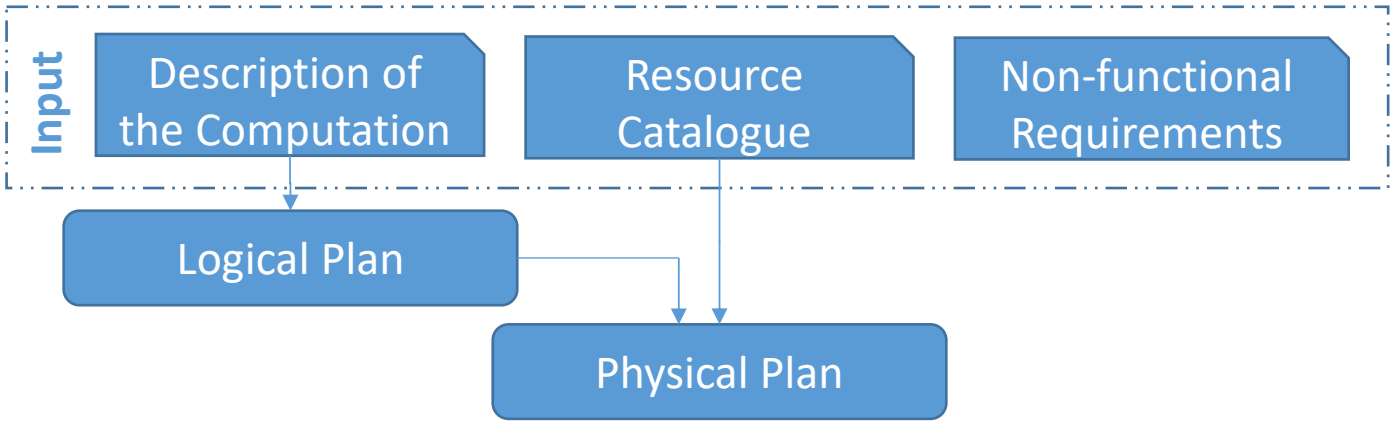
Step count algorithm[1] in EPL

1. INSERT INTO AccelEvent
   SELECT getAccelData(25, 60) FROM AccelEventSource
2. INSERT INTO EdEvent
   SELECT Math.pow(x*x+y*y+z*z, 0.5) AS ed, ts
   FROM AccelEvent WHERE vibe=0
3. INSERT INTO StepEvent
   SELECT ed1('ts') as ts FROM *EdEvent*
   MATCH RECOGNIZE (MEASURES A AS ed1, B AS ed2 PATTERN (A B)
   DEFINE A AS (A.ed > THR), B AS (B.ed ≤ THR))
4. INSERT INTO StepCount SELECT count(*) as steps FROM
   StepEvent.win:time_batch(120 sec)
5. SELECT persistResult(steps, "time_series", "step_sum") FROM StepCount



[1] N. Zhao, "Full-featured pedometer design realized with 3-axis digital accelerometer," *Analog Dialogue*, vol. 44, no. 06, 2010.

PATHfinder: Optimisation

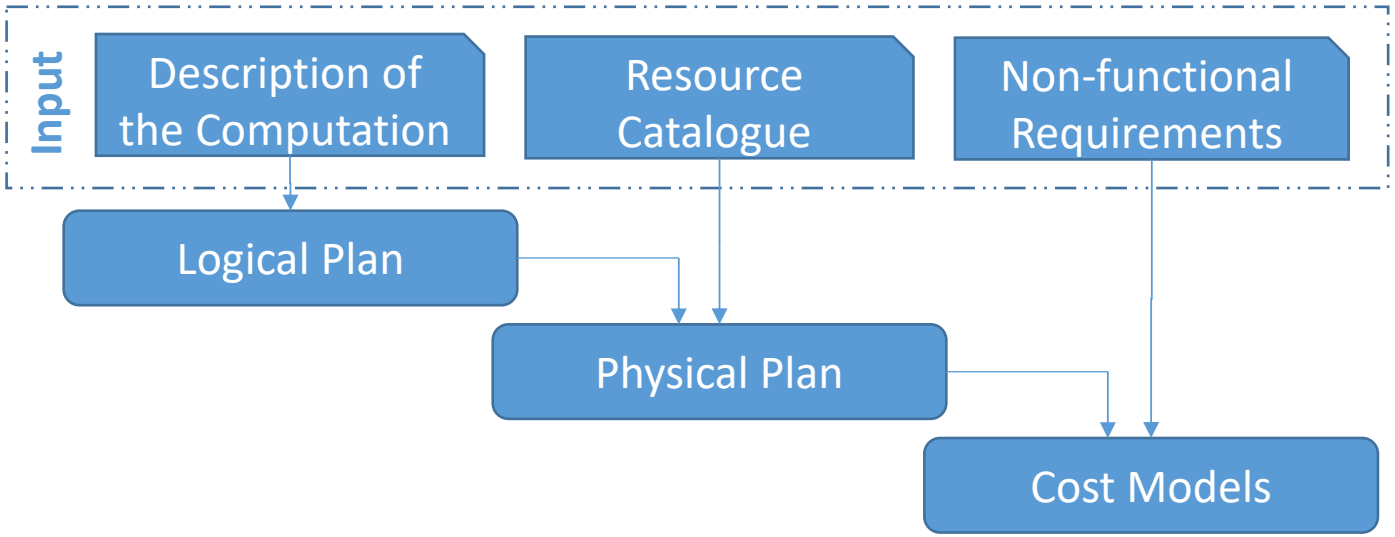


PP_0	~~PP_1~~	~~PP_2~~	PP_3	~~PP_4~~	~~PP_5~~
PP_6	PP_7	PP_8	~~PP_9~~	PP_{10}	~~PP_{11}~~
PP_{12}	~~PP_{13}~~	~~PP_{14}~~	PP_{15}	~~PP_{16}~~	PP_{17}
~~PP_{18}~~	~~PP_{19}~~	PP_{20}	~~PP_{21}~~	...	PP_{225}

- Logical Optimisation[2]
  - Pushing Selects & Windows closer to the data source
- Physical Optimisation
  - Enumerating Physical Plans
  - Placement of the physical plans on available infrastructure
  - 225 Physical Plans
- Physical Plan Pruning
  - Removing non-deployable plans based on infrastructure capabilities
  - 18 Physical Plans

[2] Hirzel, Martin, Robert Soulé, Scott Schneider, Buğra Gedik, and Robert Grimm. 'A Catalog of Stream Processing Optimizations'. ACM Computing Surveys 46

PATHfinder: Energy Cost Model

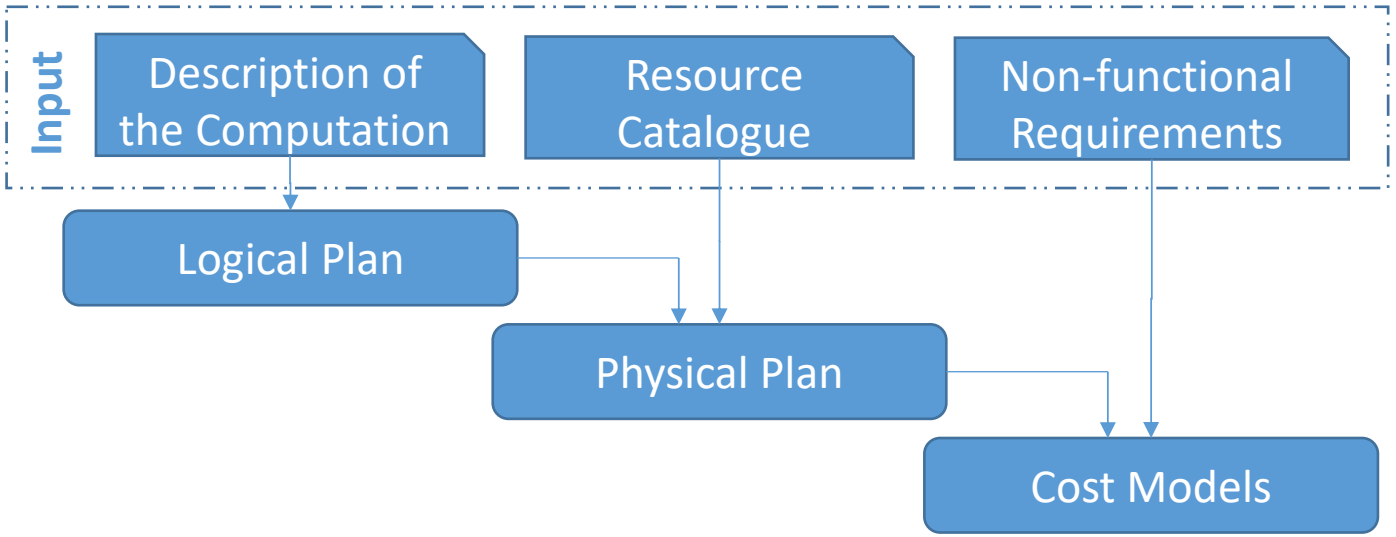


$$EI = OS_{idle} + \sum_{i}^{n} comp\_cost_i + \frac{msg\_count * net\_cost + BLE_{active} * BLE_{duration}}{cycle\_length}$$

- Cost Models
  - Energy cost model[3]
  - Power coefficients with confidence Intervals
  - Estimated Battery Life

[3] M. Forshaw, N. Thomas, and A. S. McGough, "The case for energy- aware simulation and modelling of internet of things (iot)," *ACM ENERGY-SIM*, 2016.

PATHfinder : Energy Cost Model



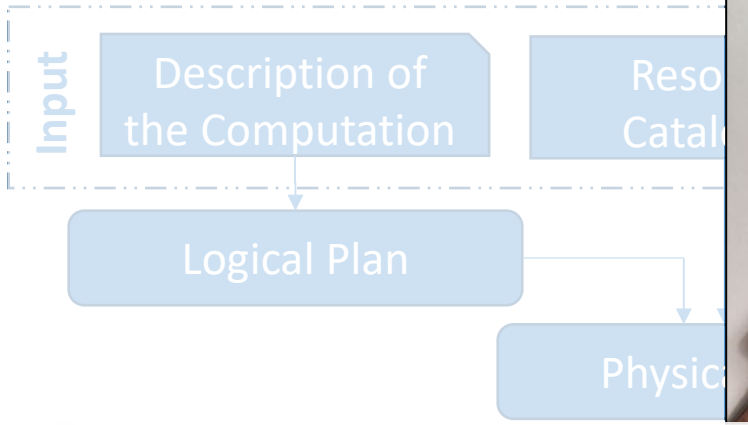
$$EI = \underbrace{OS_{idle} + \sum_{i}^{n} comp\_cost_i}_{} + \frac{msg\_count * net\_cost + BLE_{active} * BLE_{duration}}{cycle\_length}$$

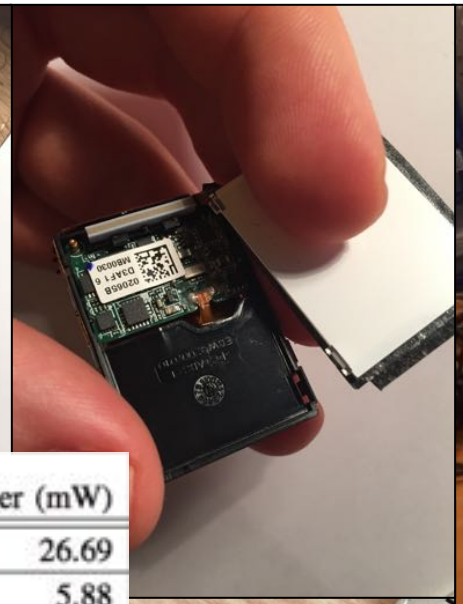
- Cost Models
  - Energy cost model
  - Power coefficients with confidence Intervals
  - Estimated Battery Life

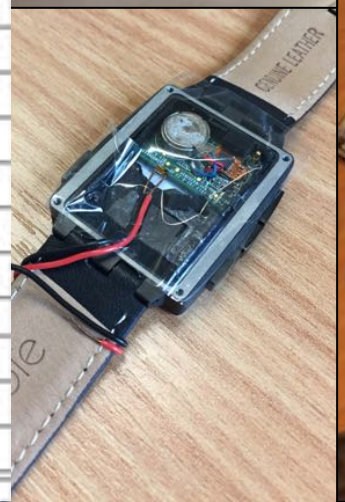
TABLE I: Power Consumption Coefficients.

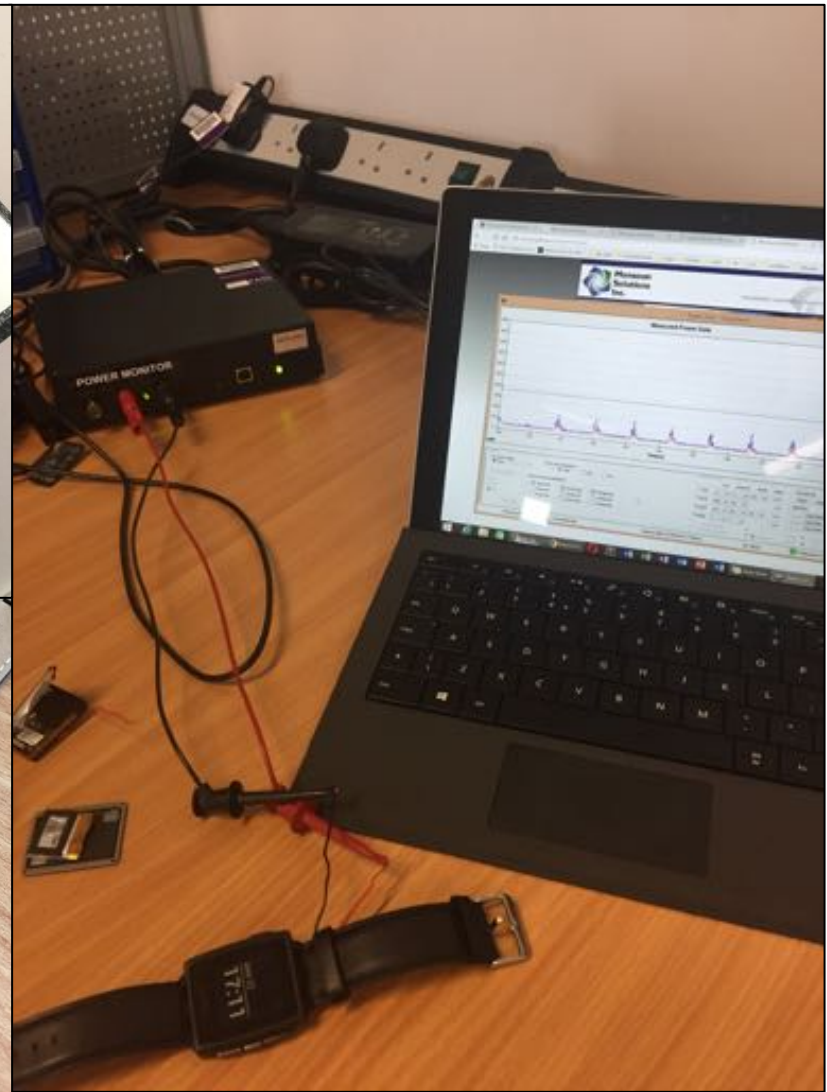
Operation	Energy Impact (mJ)	Conf Int
OS_{idle}	1.78	± 0.0370
25 Hz sampling	0.06	± 0.0153
SELECT	0.09	± 0.0416
ED	0.34	± 0.0665
POW	0.03	± 0.1039
WIN	0.06	± 0.0605
net_cost	5.06	± 0.2747
BLE_{active}	12.12	

PATHfinder : Energy Cost Model



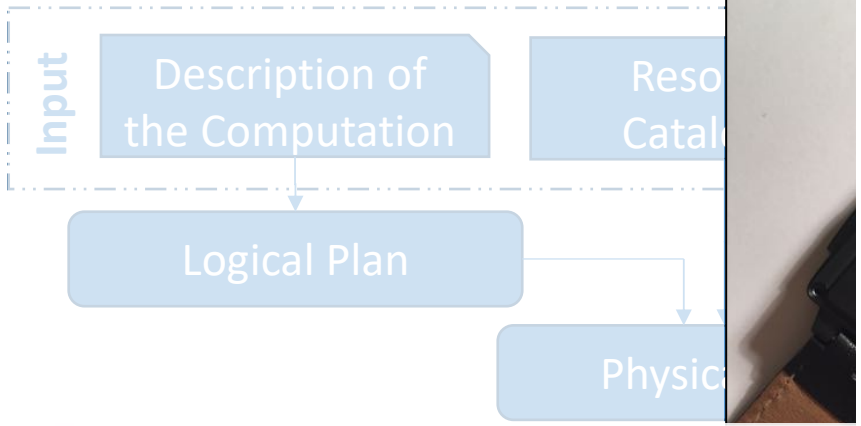




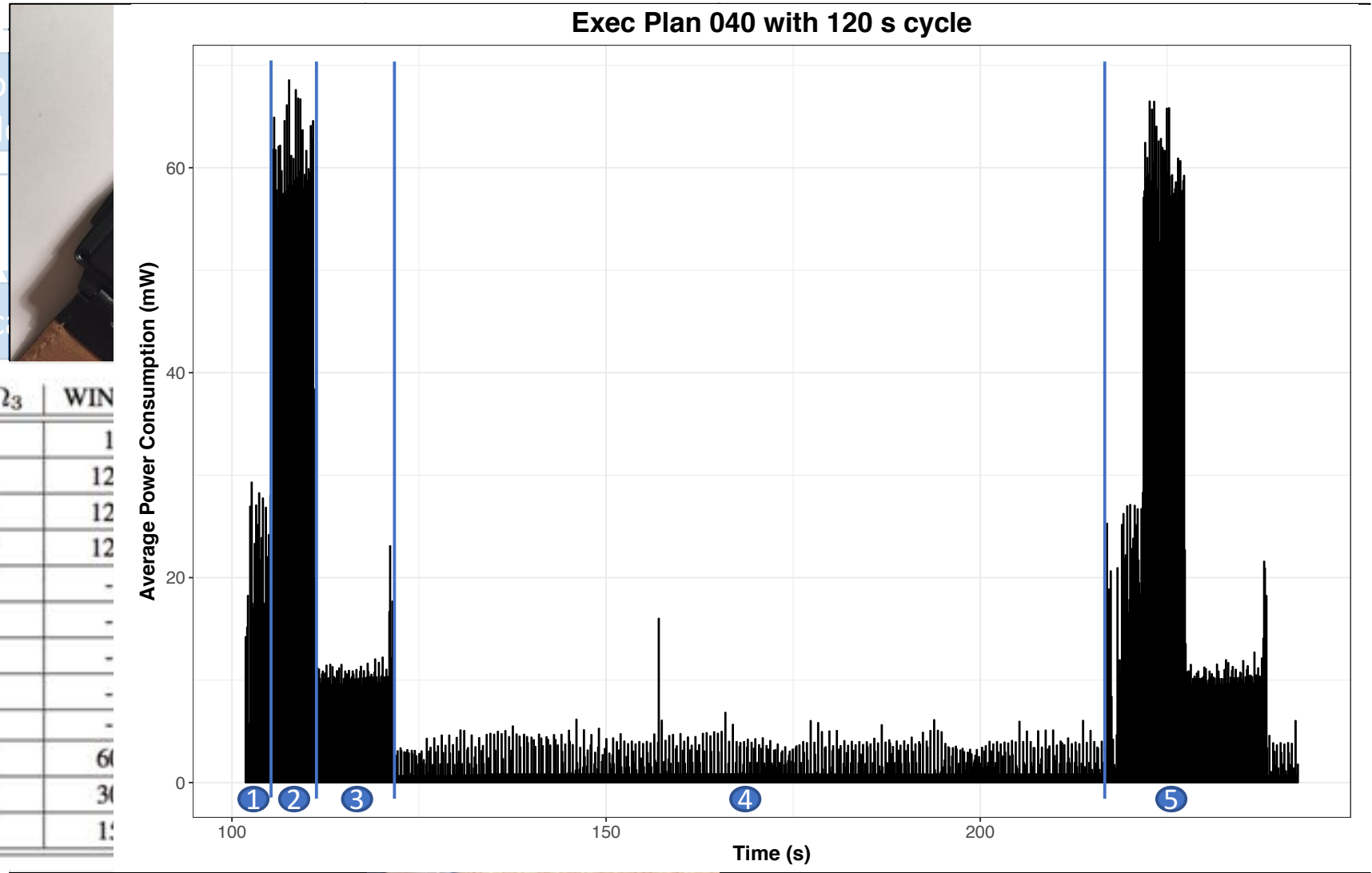


ID	Data Ω_1	SELECT σ_1	ED Ω_2	POW Ω_3	WIN ω_1	# msgs	Power (mW)
037	✓	-	-	-	1	2.5	26.69
060	✓	✓	✓	-	120	60	5.88
040	✓	✓	✓	✓	120	60	5.97
041	✓	✓	✓	✓	120	0	2.37
042	✓	✓	✓	✓	-	0	2.31
043	✓	✓	✓	-	-	0	2.27
044	✓	✓	-	-	-	0	1.96
045	✓	-	-	-	-	0	1.84
046	-	-	-	-	-	0	1.78
050	✓	✓	✓	✓	60	30	7.15
051	✓	✓	✓	✓	30	15	9.63
052	✓	✓	✓	✓	15	7.5	13.41

PATHfinder : Energy Cost Model

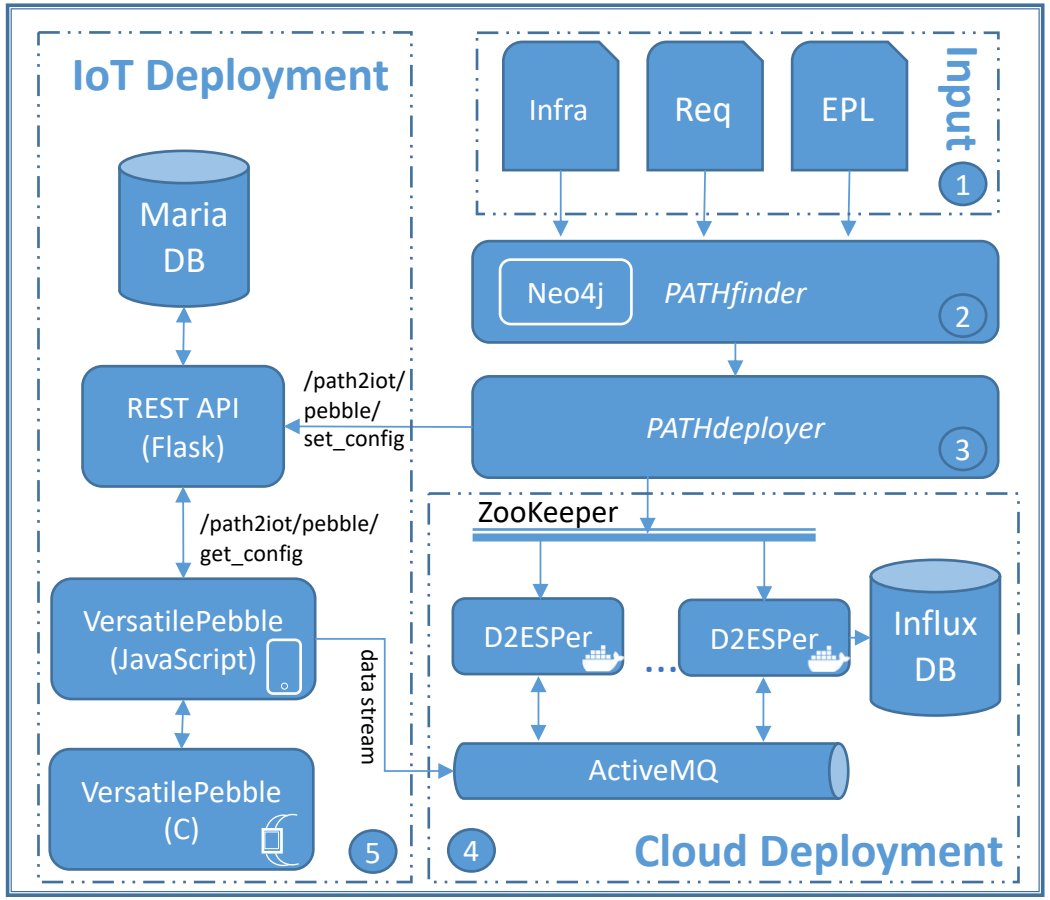


ID	Data Ω_1	SELECT σ_1	ED Ω_2	POW Ω_3	WIN
037	✓	-	-	-	1
060	✓	✓	✓	-	12
040	✓	✓	✓	✓	12
041	✓	✓	✓	✓	12
042	✓	✓	✓	✓	-
043	✓	✓	✓	-	-
044	✓	✓	-	-	-
045	✓	-	-	-	-
046	-	-	-	-	-
050	✓	✓	✓	✓	6(
051	✓	✓	✓	✓	3(
052	✓	✓	✓	✓	1:



PATHdeployer : Architecture Overview

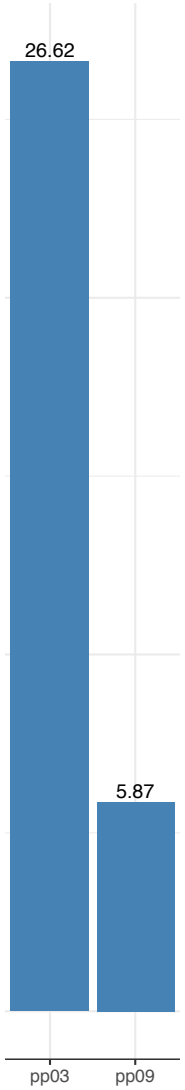
- Cloud Deployment
  - **ZooKeeper**: configuration delivery
  - **ActiveMQ**: event propagation
  - **D2ESPer**: in-house built dynamic ESPer based stream processing tool
  - **InfluxDB**: time series database
- IoT Deployment
  - **Flask** REST API: configuration delivery for IoT devices
  - **MariaDB**: storage endpoint
  - **IoT agents** – iPhone, Pebble Watch



Results

Plan	watch	phone	cloud	Energy Impact (mJ)
pp00	$\Omega_1 \sigma_1 \Omega_2 \Omega_3$	sxfer	$\Omega_4 \omega_1 \Omega_6 \Omega_5$	27.08
pp01	$\Omega_1 \sigma_1 \Omega_2$	sxfer	$\Omega_3 \Omega_4 \omega_1 \Omega_6 \Omega_5$	27.05
pp02	$\Omega_1 \sigma_1$	sxfer	$\Omega_2 \Omega_3 \Omega_4 \omega_1 \Omega_6 \Omega_5$	26.71
pp03	Ω_1	sxfer	$\sigma_1 \Omega_2 \Omega_3 \Omega_4 \omega_1 \Omega_6 \Omega_5$	baseline 26.62
pp04	$\Omega_1 \sigma_1 \Omega_2 \Omega_3 \omega_1$	sxfer	$\Omega_4 \Omega_6 \Omega_5$	5.91
pp05	$\Omega_1 \sigma_1 \Omega_2 \Omega_3$	sxfer	$\omega_1 \Omega_4 \Omega_6 \Omega_5$	27.08
pp06	$\Omega_1 \sigma_1 \Omega_2$	sxfer	$\Omega_3 \omega_1 \Omega_4 \Omega_6 \Omega_5$	27.05
pp07	$\Omega_1 \sigma_1$	sxfer	$\Omega_2 \Omega_3 \omega_1 \Omega_4 \Omega_6 \Omega_5$	26.71
pp08	Ω_1	sxfer	$\sigma_1 \Omega_2 \Omega_3 \omega_1 \Omega_4 \Omega_6 \Omega_5$	26.62
pp09	$\Omega_1 \sigma_1 \Omega_2 \omega_1$	sxfer	$\Omega_3 \Omega_4 \Omega_6 \Omega_5$	best plan 5.87
pp10	$\Omega_1 \sigma_1 \Omega_2$	sxfer	$\omega_1 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	27.05
pp11	$\Omega_1 \sigma_1$	sxfer	$\Omega_2 \omega_1 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	26.71
pp12	Ω_1	sxfer	$\sigma_1 \Omega_2 \omega_1 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	26.62
pp13	$\Omega_1 \sigma_1 \omega_1$	sxfer	$\Omega_2 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	10.6
pp14	$\Omega_1 \sigma_1$	sxfer	$\omega_1 \Omega_2 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	26.71
pp15	Ω_1	sxfer	$\sigma_1 \omega_1 \Omega_2 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	26.62
pp16	$\Omega_1 \omega_1$	sxfer	$\sigma_1 \Omega_2 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	10.51
pp17	Ω_1	sxfer	$\omega_1 \sigma_1 \Omega_2 \Omega_3 \Omega_4 \Omega_6 \Omega_5$	26.62

Results



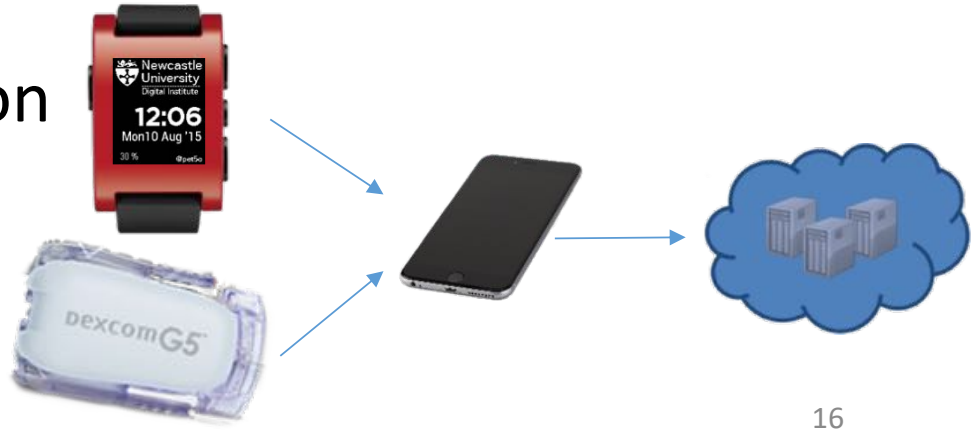
Plan	watch	phone	cloud	EI (mJ)	95 % conf int	Power (mW)	Bat Life (h)
pp03	Ω_1	sxfer	$\sigma_1 \Omega_2 \Omega_3 \Omega_4 \omega_1 \Omega_6 \Omega_5$	5.87	5.70 - 6.05	5.88	18.0-18.2
pp09	$\Omega_1 \sigma_1 \Omega_2 \omega_1$	sxfer	$\Omega_3 \Omega_4 \Omega_6 \Omega_5$	26.62	26.48 - 26.76	26.69	79.5-84.4

- **453 %** battery life improvement[5]
- **3x** data reduction between wearable and cloud
- Non-functional requirement satisfied

[5] P. Michalák, P. Watson, "PATH2iot: A Holistic, Distributed Stream Processing System", *CloudCom*, 2017

Holistic Distributed Stream Processing in IoT Environments

- Holistic, Distributed Stream Processing System
  - Design and open-source implementation[6]
  - EPL decomposition
  - Logical and Physical Optimisation
- Energy Impact coefficients for Pebble Watch
  - Battery life increased dramatically
- PoC Deployment Architecture
- Future work on Multi-objective optimisation
  - e.g. Bandwidth, Performance, Accuracy



[6] https://github.com/PetoMichalak/iotPower