Elastic Computing and Engineering Elastic Applications in the Cloud

Summer SOC, Crete, 2 July 2013

Schahram Dustdar and Hong-Linh Truong

Distributed Systems Group
TU Vienna

http://dsg.tuwien.ac.at/research/viecom/

Distributed Systems Group

Acknowledgements

Includes some joint work with Kamal Bhattacharya, Muhammad Z.C. Candra, Georgiana Copil, Daniel Moldovan, Mirela Riveri, Ognjen Scekic

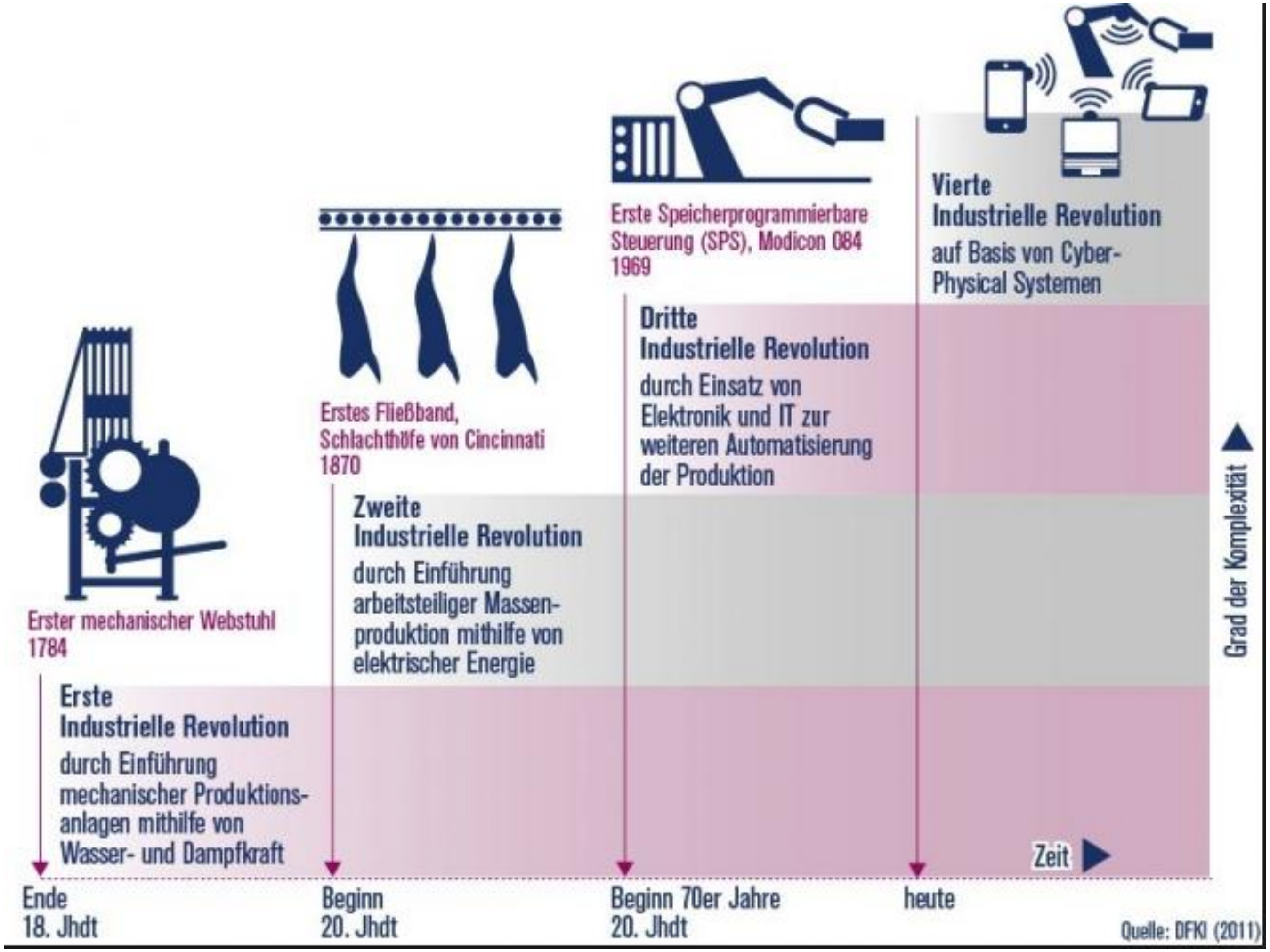
NOTE: The content includes some ongoing work

Outline

- Part 1: Elastic Computing
  - Motivation for multi-dimensional elasticity
  - Quality/cost/benefits analytics
  - HBS cloud concepts
  - Conclusions
- Part 2: Engineering Elastic Applications in the Cloud
  - Programming hybrid services for solving (in)dependent tasks
  - Programming incentives
  - Controlling and monitoring elasticity
  - Conclusions
- Part 3: Demonstration of elasticity control and monitoring

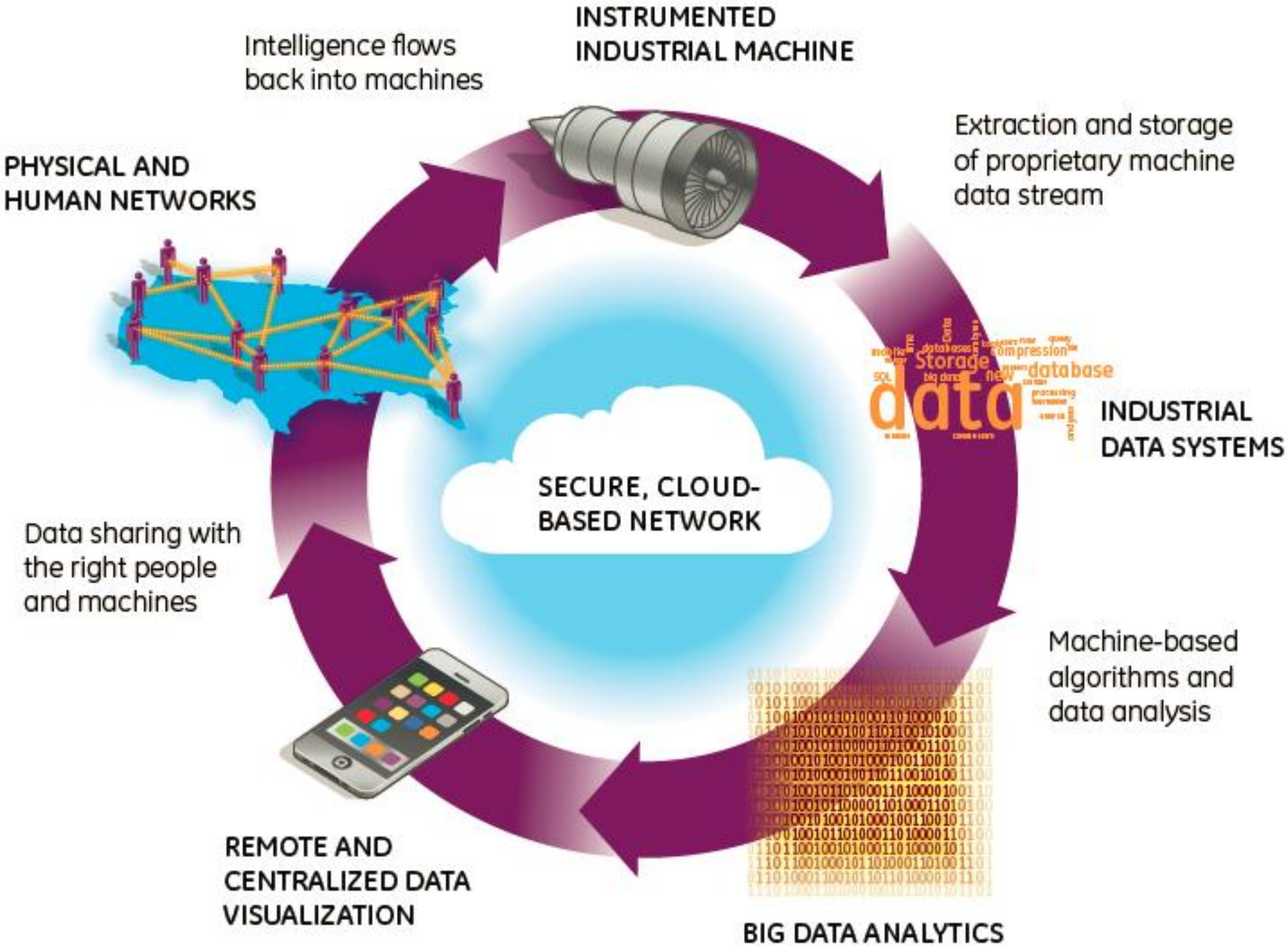
PART 1 – ELASTIC COMPUTING

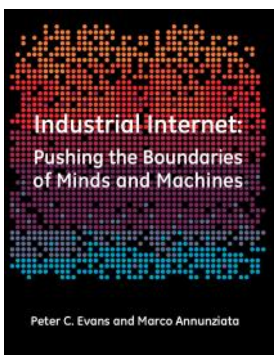
"Industrie 4.0" – German industrial CPS



VDE Dialog 2/2013, S.15

Industrial Internet





Industrial Internet –

Pushing the Boundaries of Minds and Machines, Nov. 2012

Cloud Computing?

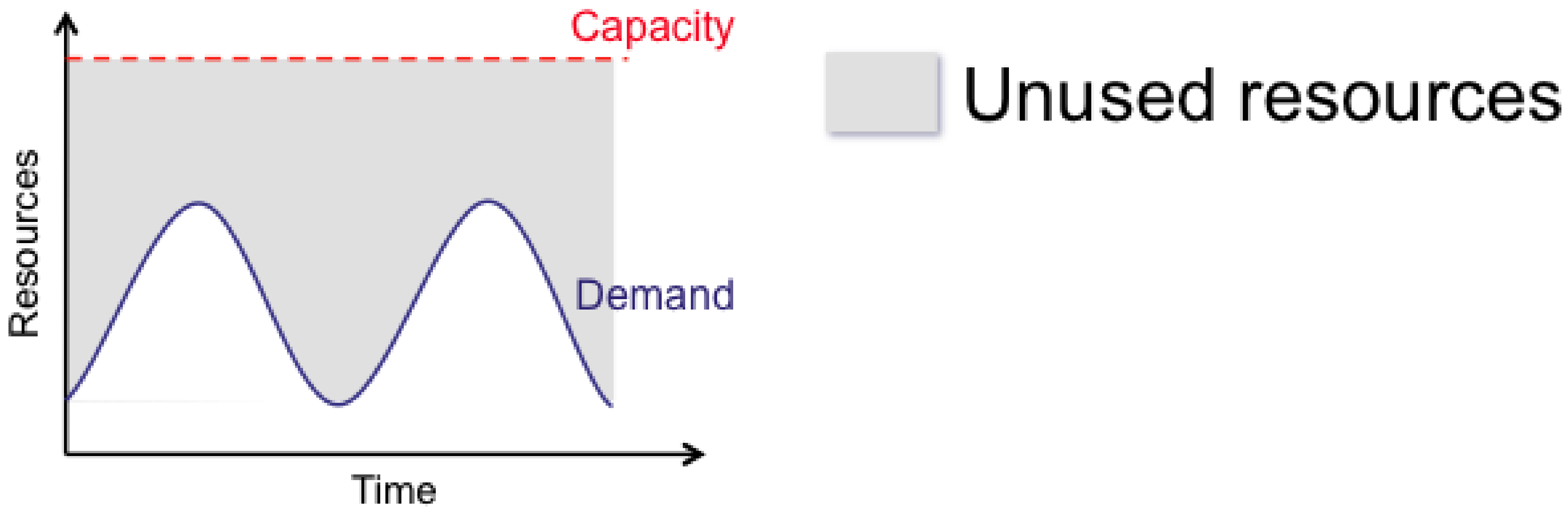
1. “Resources” provided as services

2. Illusion of infinite resources

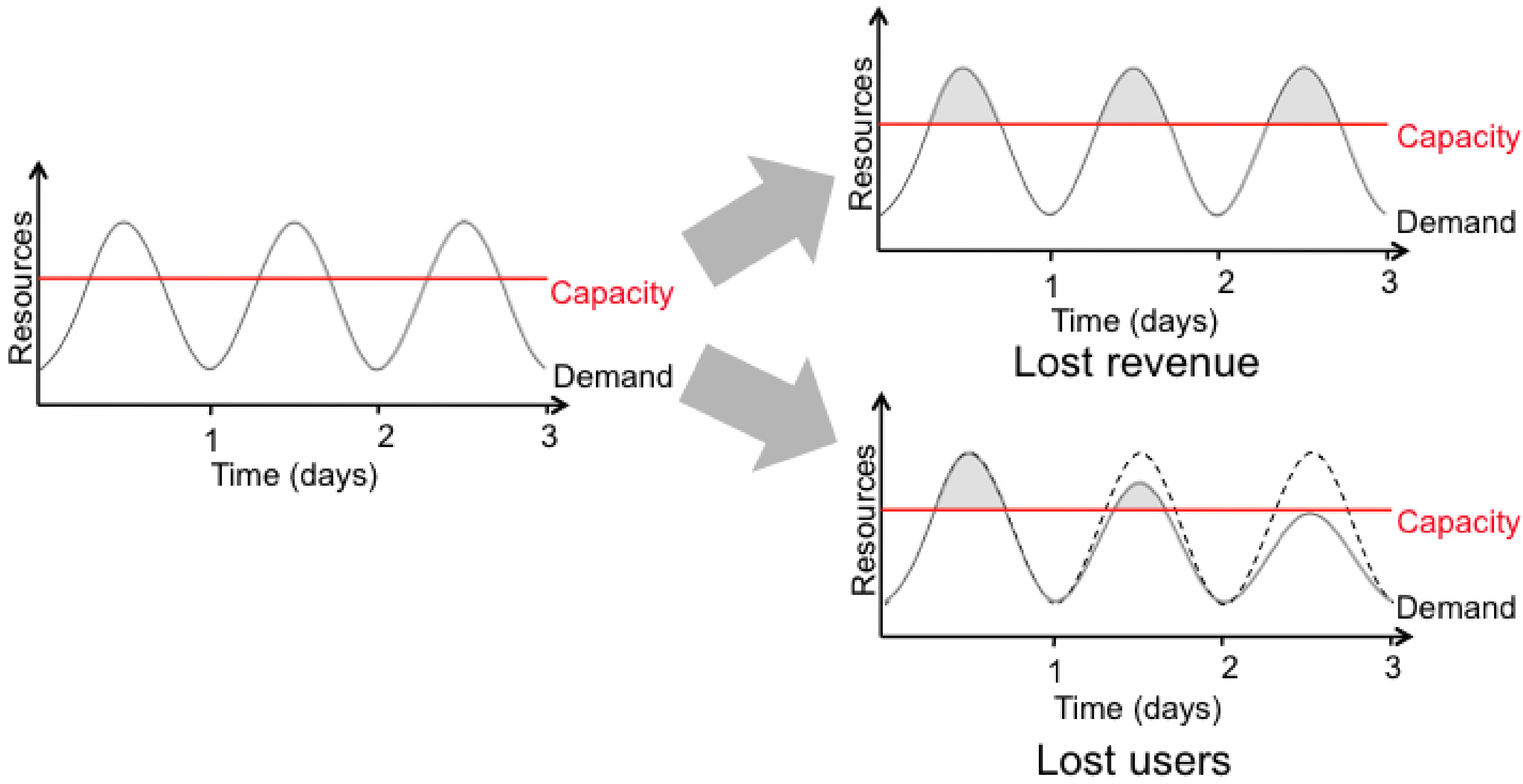
3. Usage-based pricing model ->

   New and connected business models

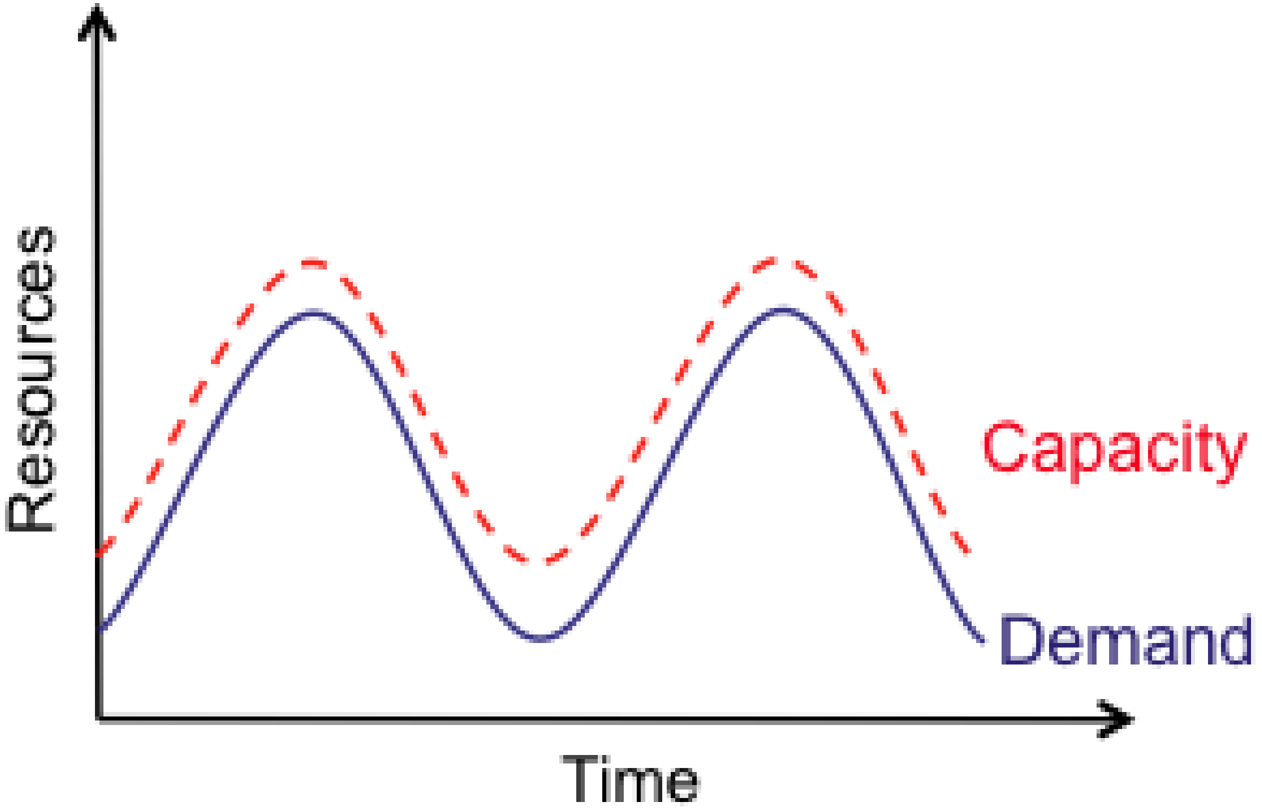
Overprovisioning – Provider perspective



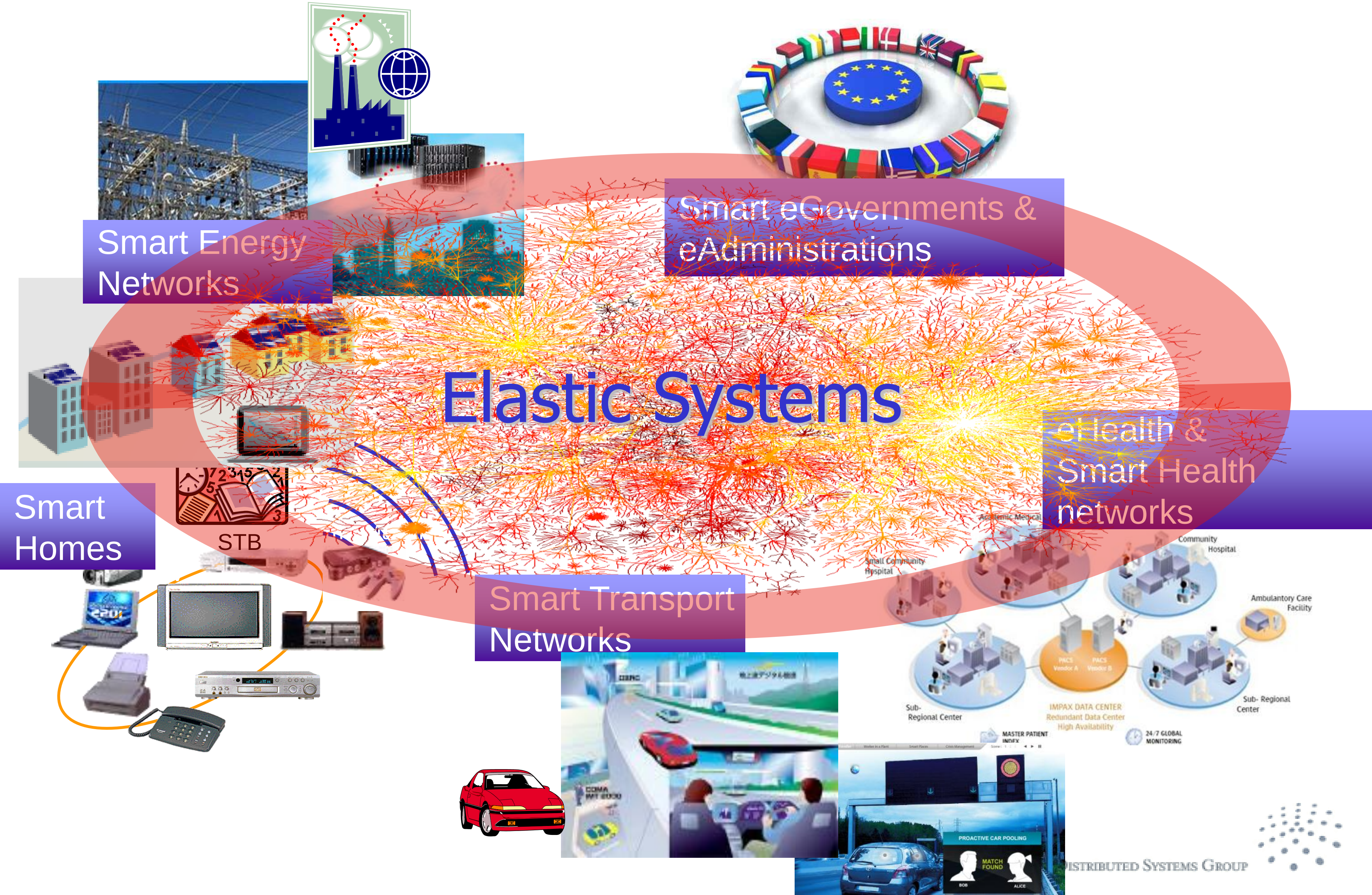
Underprovisioning – Provider perspective



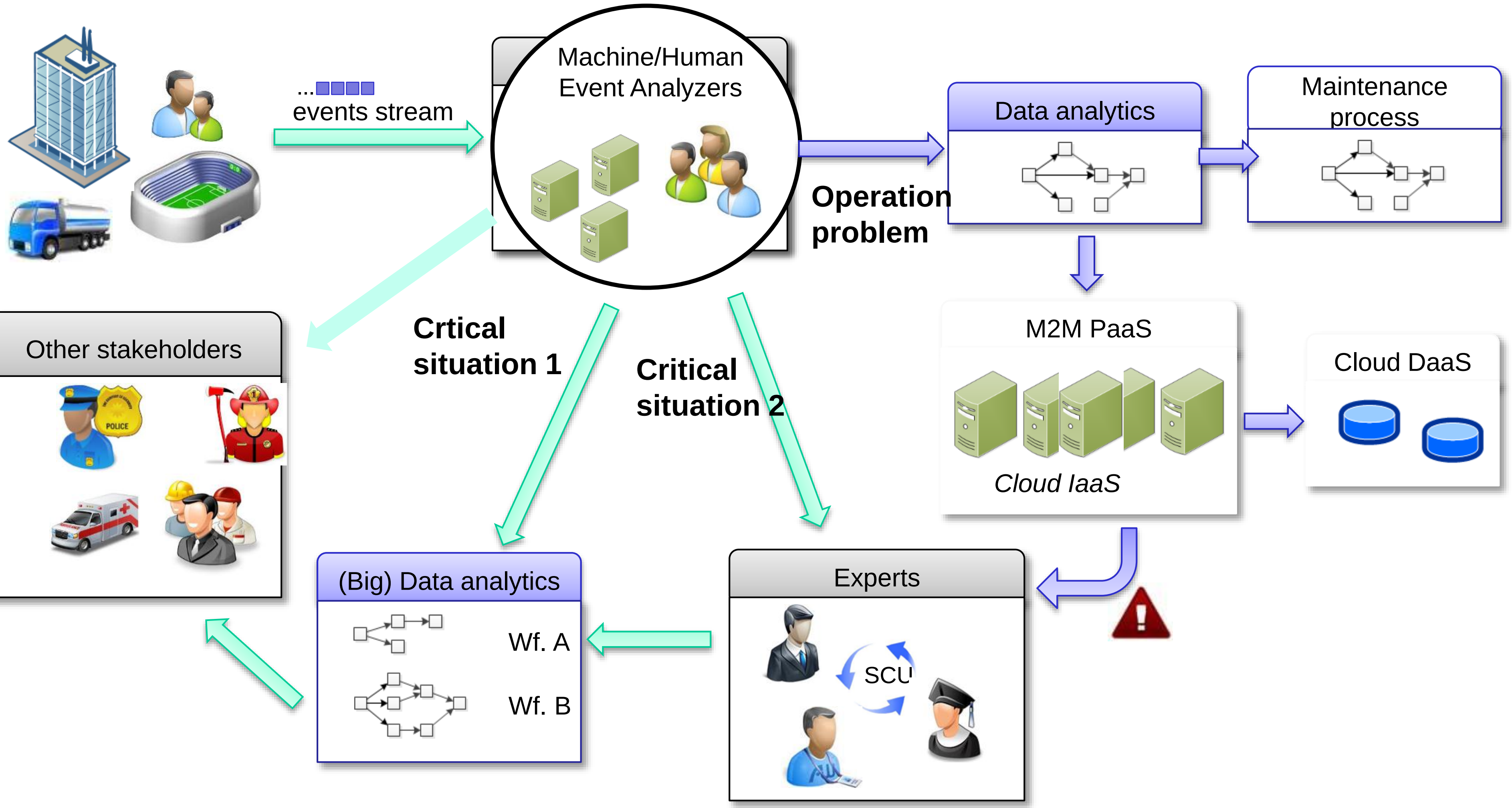
Elastic Provisioning



Smart Evolution – People, Services,Things



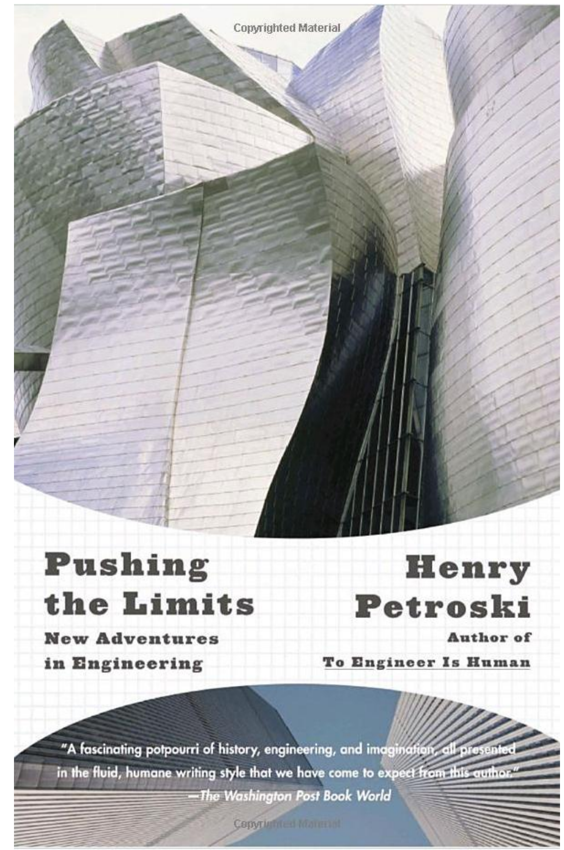
The goal: Elastic System



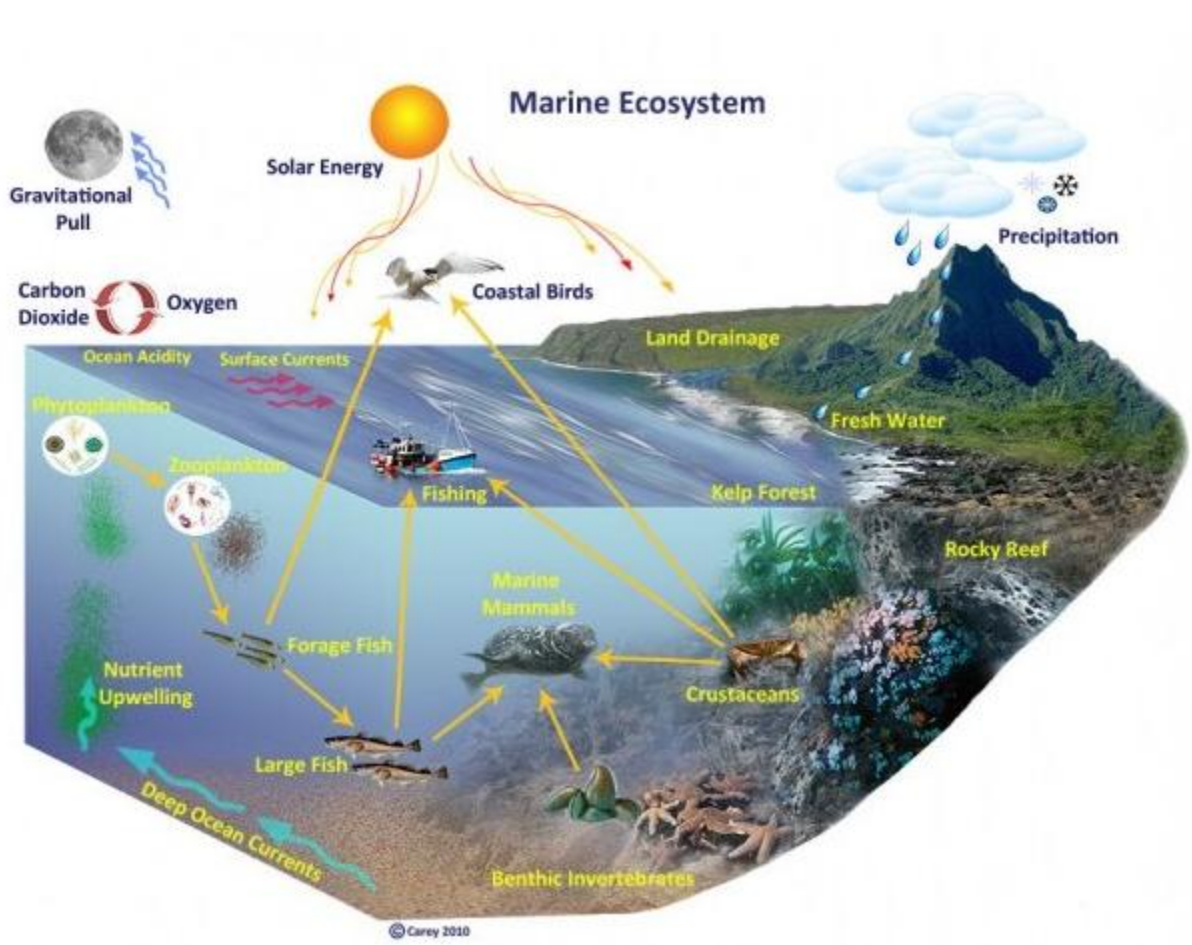
A New Perspective is Required

"The older is not always a reliable model for the newer, the smaller for the larger, or the simpler for the more complex….

Making something greater than any existing thing necessarily involves going beyond experience."



Think Ecosystems:
People, Services, Things



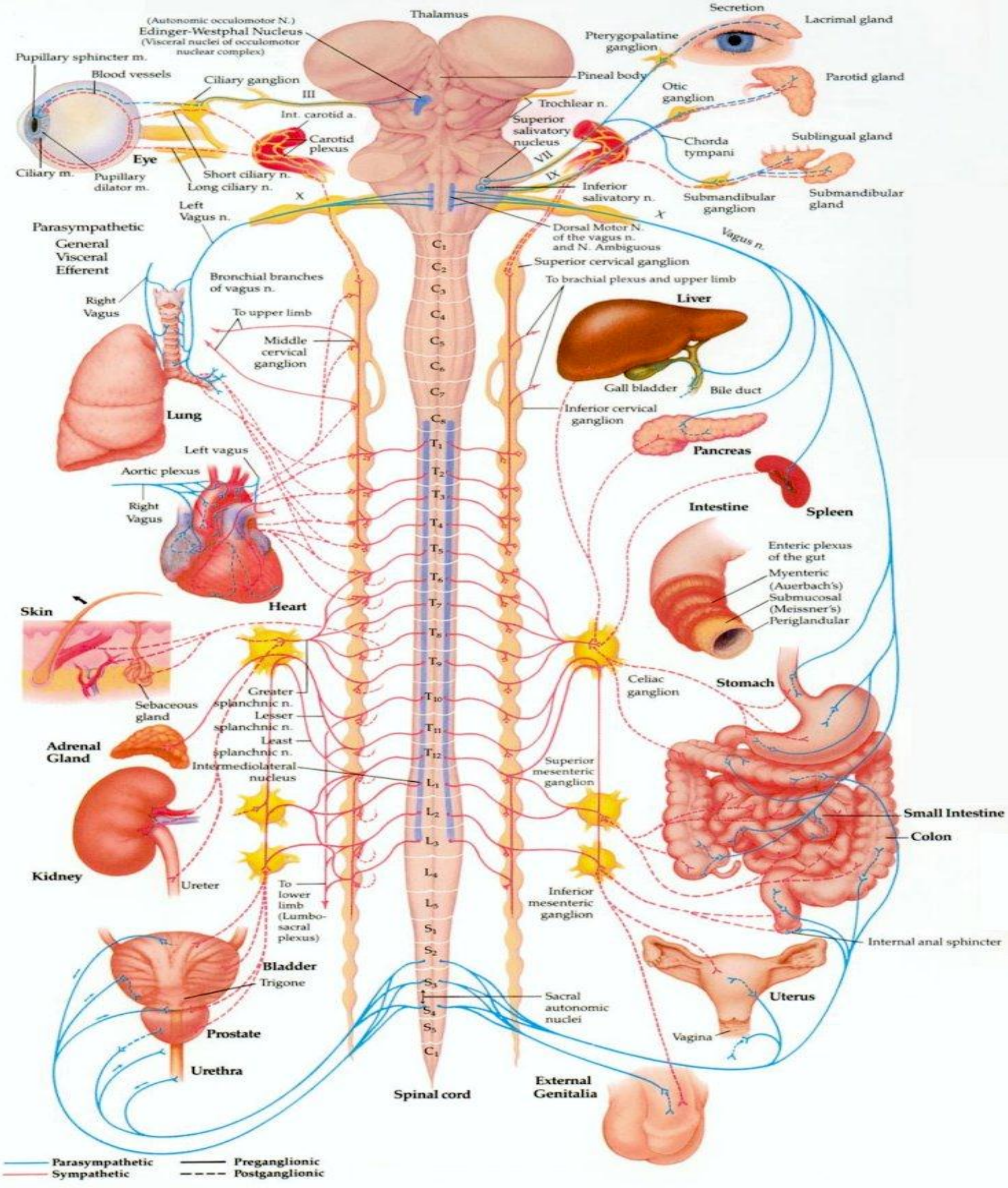
Marine Ecosystem: http://www.xbordercurrents.co.uk/wildlife/marine-ecosystem-2

Diverse users with complex networked dependencies and intrinsic adaptive behavior – has:

1. **Robustness mechanisms**: achieving stability in the presence of disruption
2. **Measures of health**: diversity, population trends, other key indicators

Evolution of **Large-Scale & Collective** Problem solving

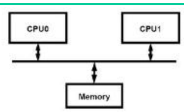
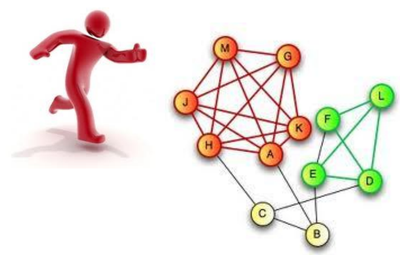
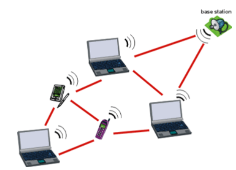
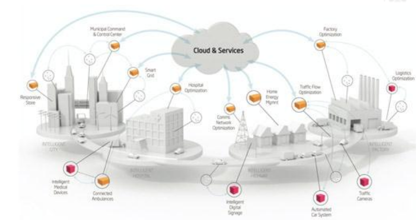
Distributed Systems Group



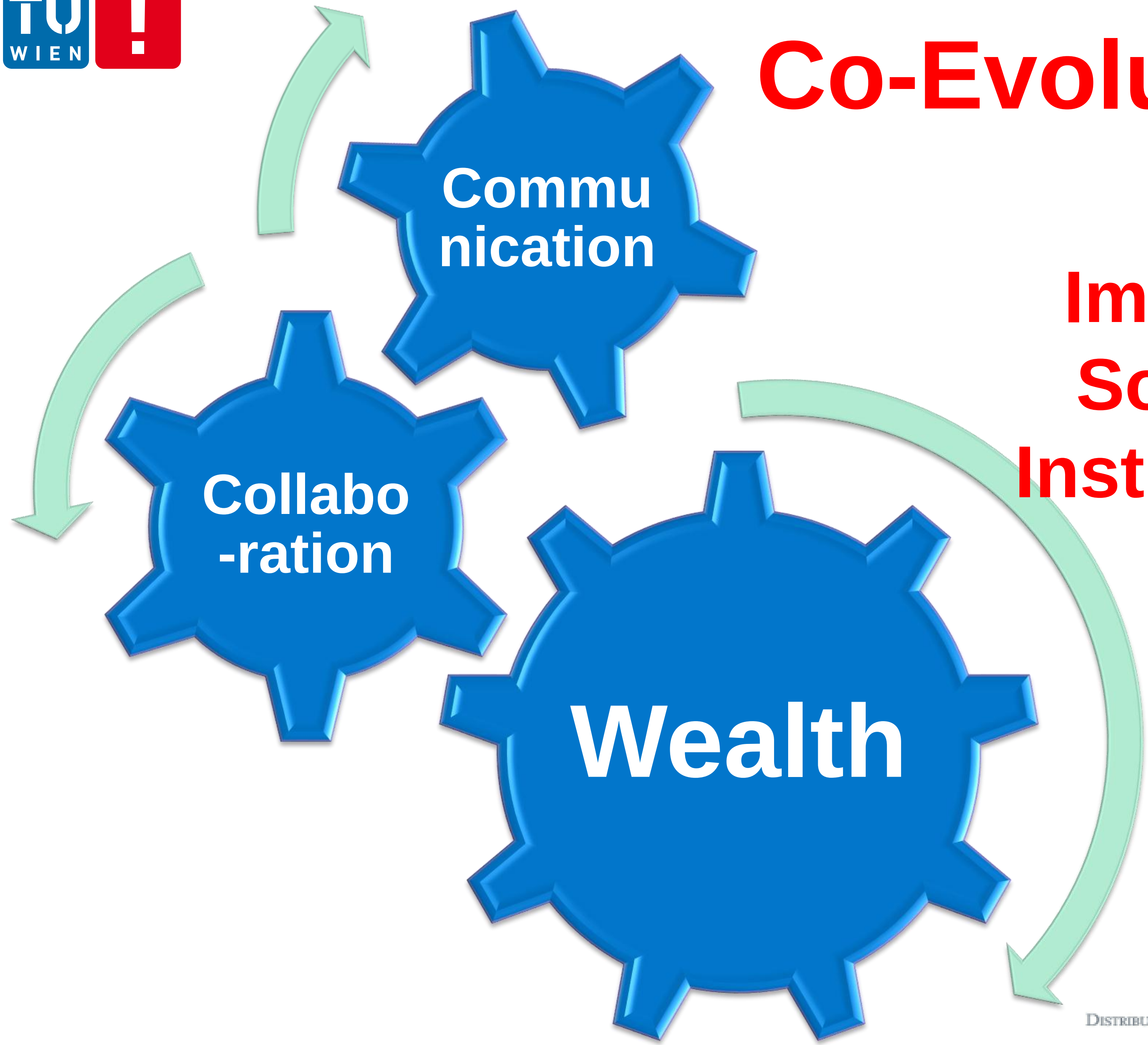
Autonomic Nervous System

Computing Models

S. Dustdar, H. Truong, “Virtualizing Software and Humans for Elastic Processes in Multiple Clouds – a Service Management Perspective”, in *International Journal of Next Generation Computing*, 2012

	Machine-based Computing	Human-based Computing	Things-based computing
Processing Unit			
Architecture	 SMP   Grid	 	Ad hoc networks  Web of things  
Comm.	 		  

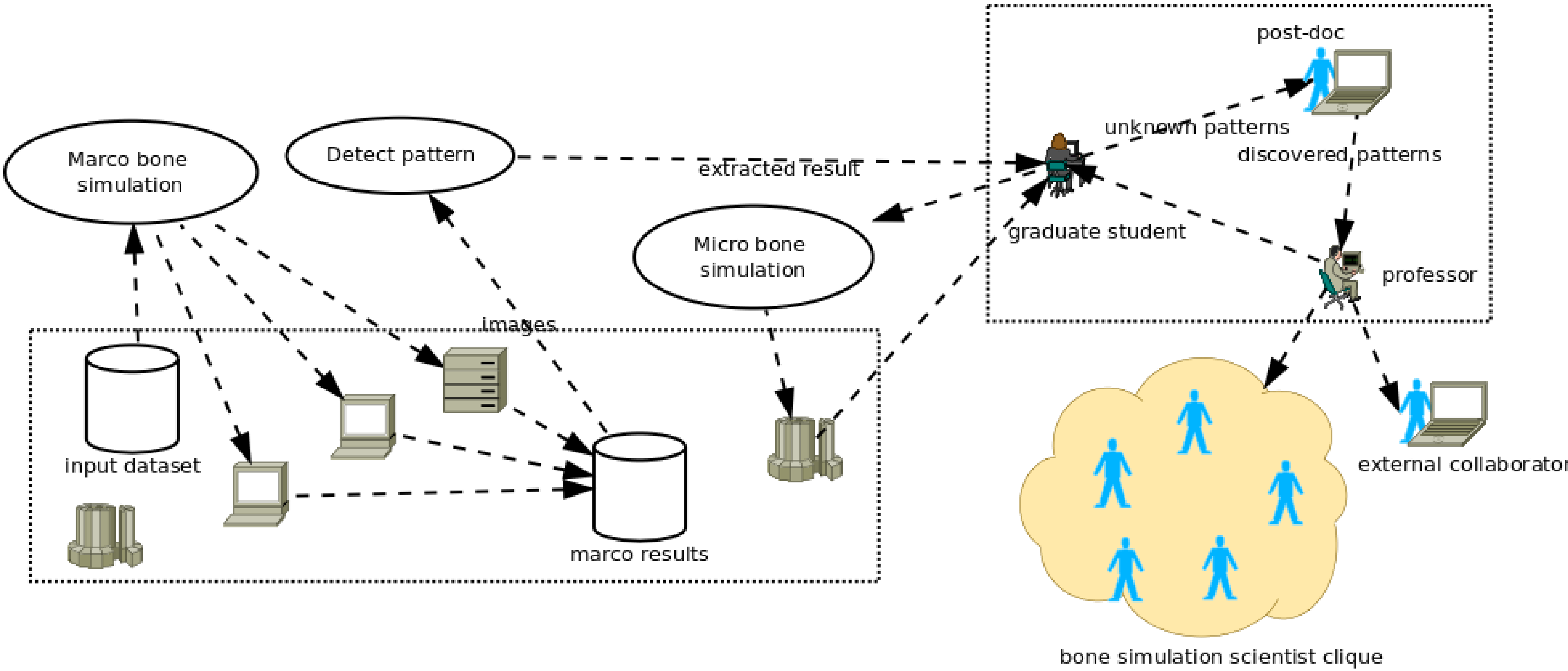
Co-Evolution



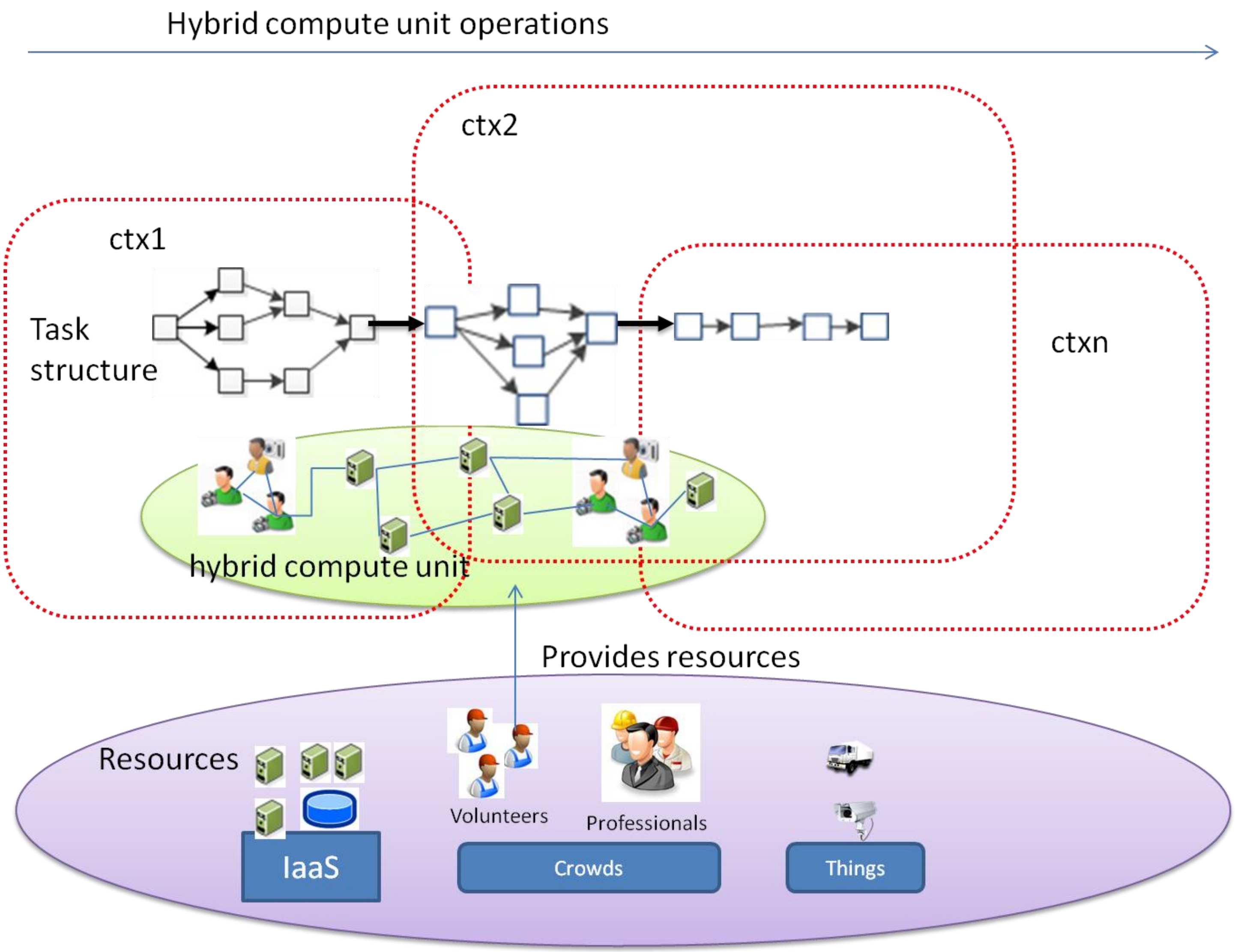
Impact on Society & Institutions

Institutions vs. Collaboration

Example: Quality evaluation in scientific multi-scale simulation



Context, Structure, and Dynamics

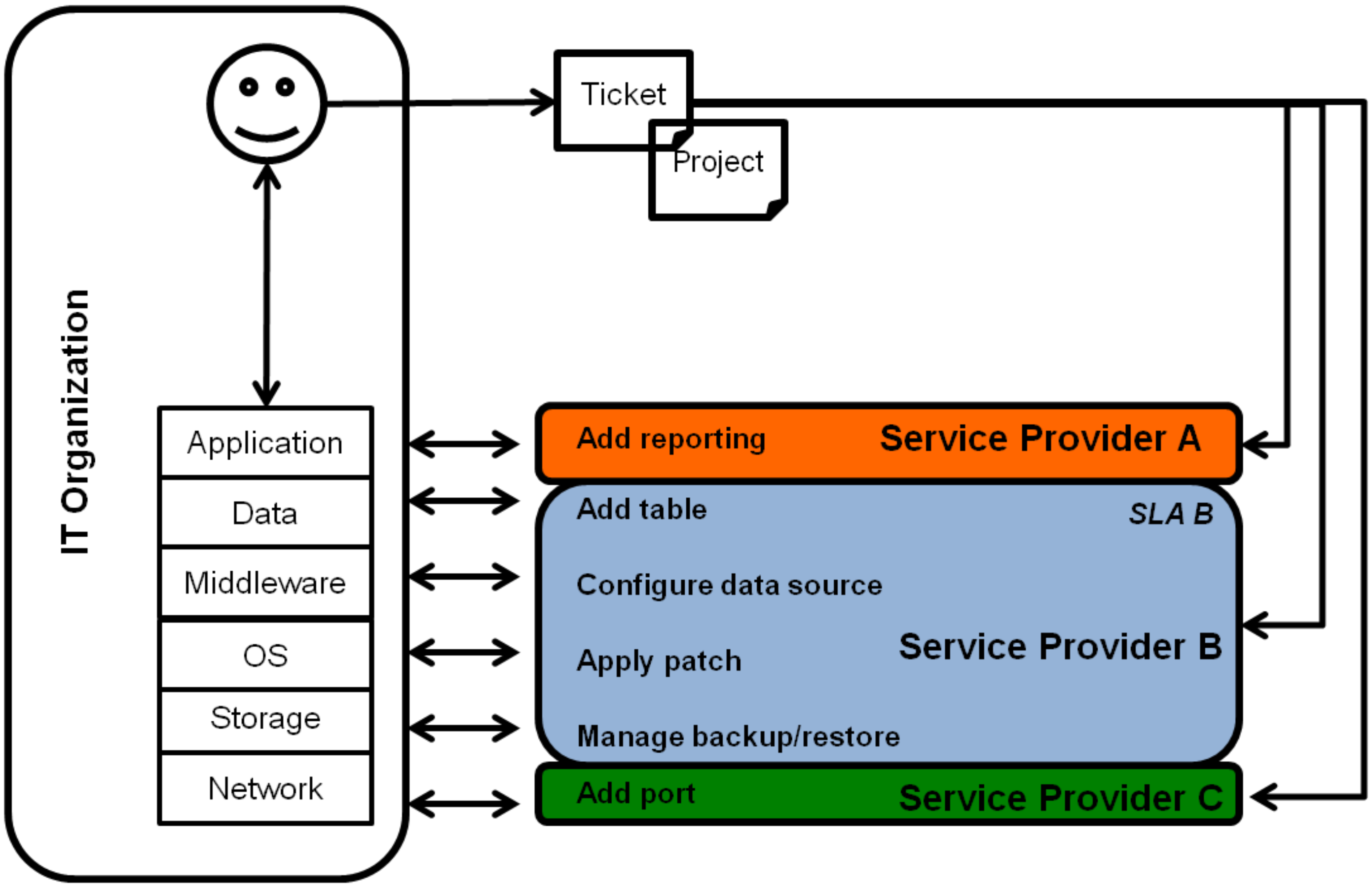


Humans as a service

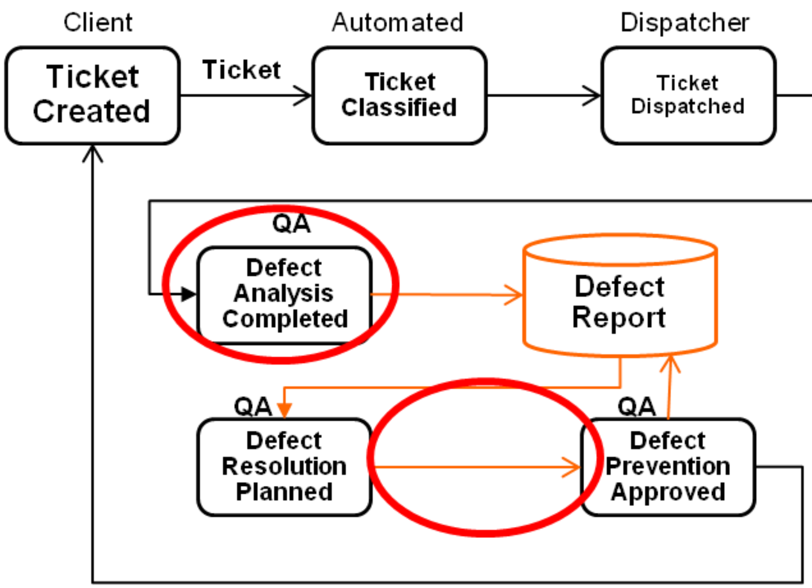
Elastic Computing – conceptualizing HBS Clouds

IT Service Delivery Factory

Tacit work


IT Organization
Ticket
Project
Application
Data
Middleware
OS
Storage
Network
Add reporting
Service Provider A
Add table
SLA B
Configure data source
Apply patch
Service Provider B
Manage backup/restore
Add port
Service Provider C

Problem statement



Area's in red circles indicate work that requires insights into:

(a) the system,
(b) how others have resolved issues
(c) how defects could be prevented at a larger scale

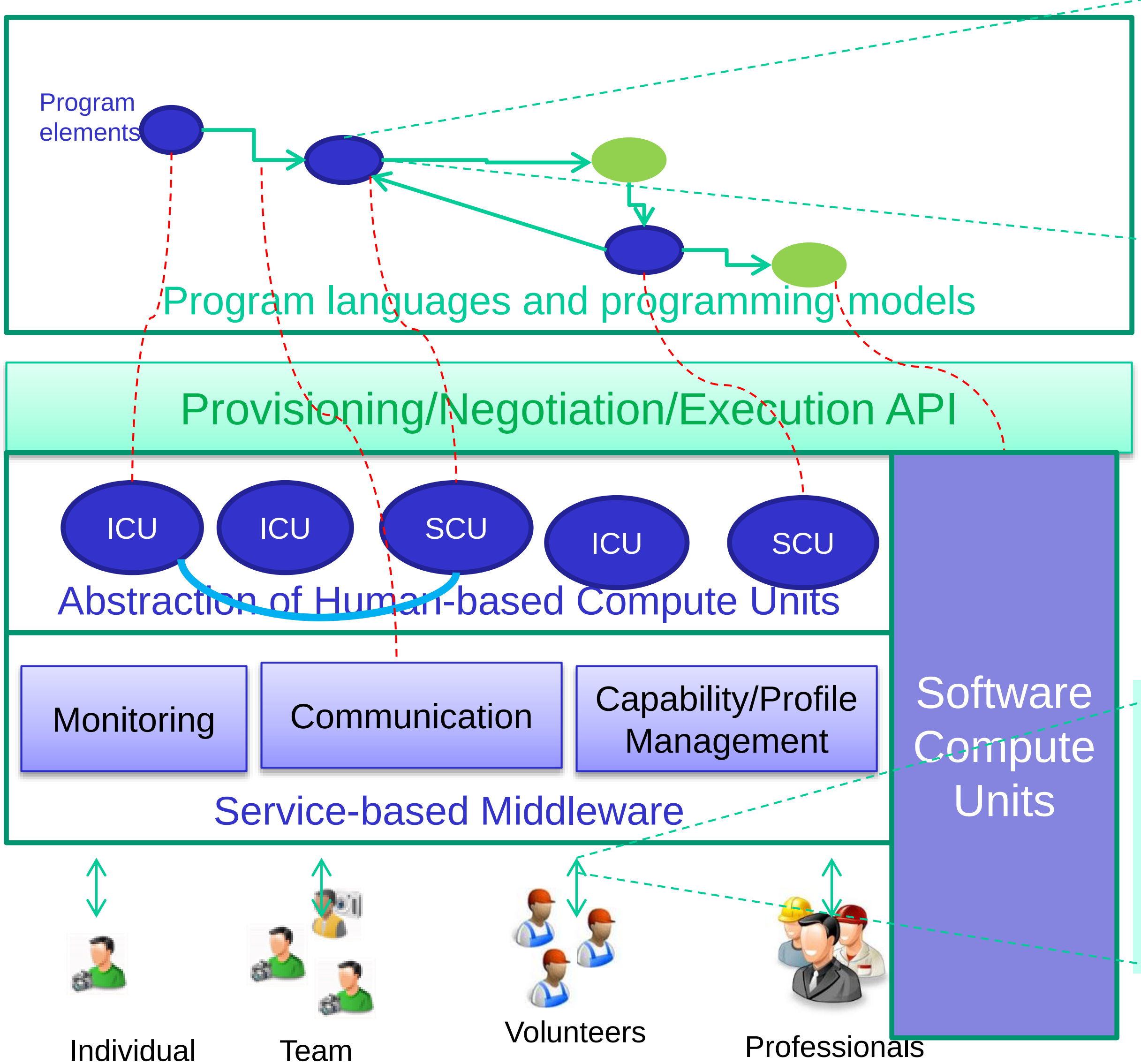
- Within the scope of transactional work, BPM is not suitable to structure the type of work that is required to be done
- The variability of underlying systems and potential actions to be taken further prevent a structured BPM-type approach

Virtualizing individual and team capabilities

- Individual capabilities and team capabilities can be virtualized using the service model
  - Individual Compute Unit (ICU) for individuals
  - Social Compute Unit (SCU) for teams

- Apply service unit models for ICU/SCU to support seamless integration between human-based computing elements and software-based computing elements

Humans in a programming paradigm



- program human actions and dependencies
- program incentive condition and rewarding action
- program result evaluation method

Hong-Linh Truong, Schahram Dustdar, Kamal Bhattacharya "**Programming Hybrid Services in the Cloud**", 10th International Conference on Service-oriented Computing (ICSOC 2012), November 12-16, 2012, Shanghai, China. Best Paper Award.

Human-to-middleware interfaces:
- visualization of collective tasks
- embedding of common forms
- mobile app

Incentives and Rewards

Incentives

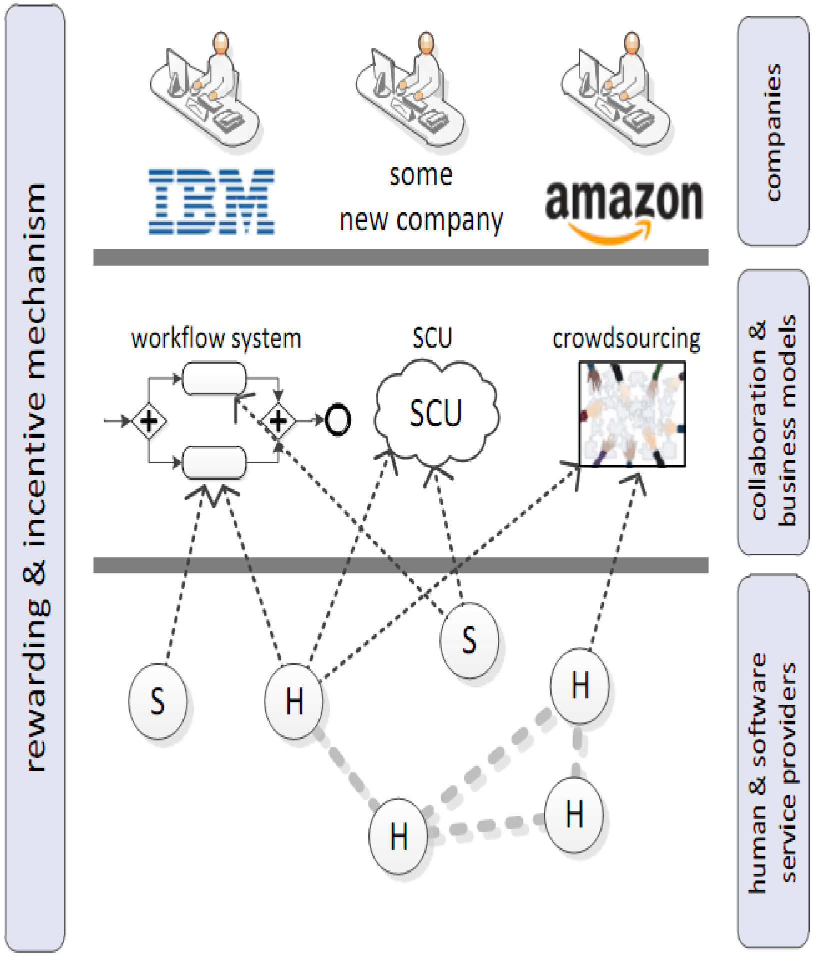
Stimulate (motivate) or discourage certain worker activities before the actual execution of those activities.

Rewards

Any kind of compensation for services rendered or retribution for wrongdoing exerted upon workers after the completion of activity.

Incentive Mechanism

A <u>set of rules</u> for assigning rewards and applying incentives.



Incentives for humans

- We need **incentive programming features**
  - Incentive strategies being composed from general components
  - Integrated into human-based service (HBS) clouds

Ognjen Scekic, Hong-Linh Truong, and Schahram Dustdar. 2013. Incentives and rewarding in social computing. Commun. ACM 56, 6 (June 2013), 72-82.

Figure 2. Incentive strategies consist of smaller, easily modeled components.

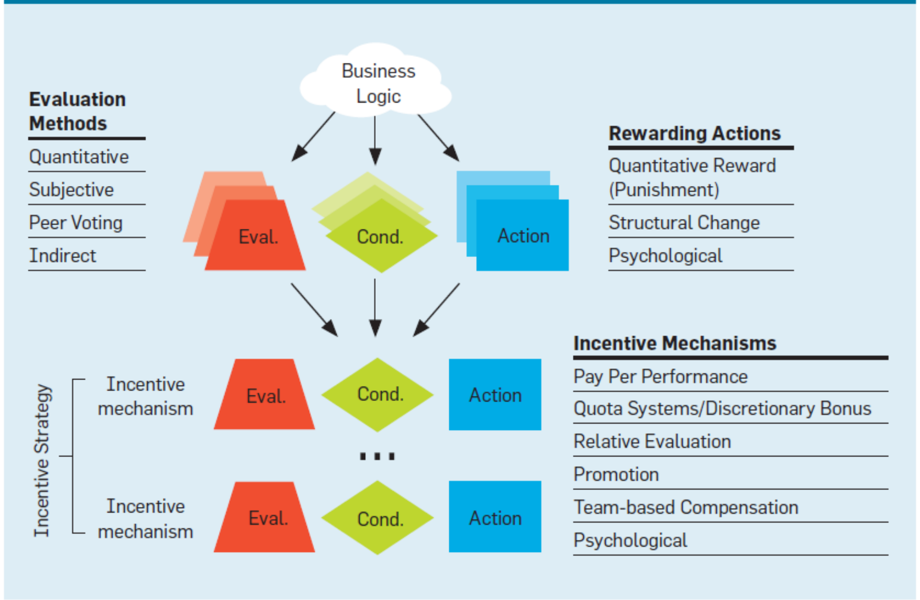
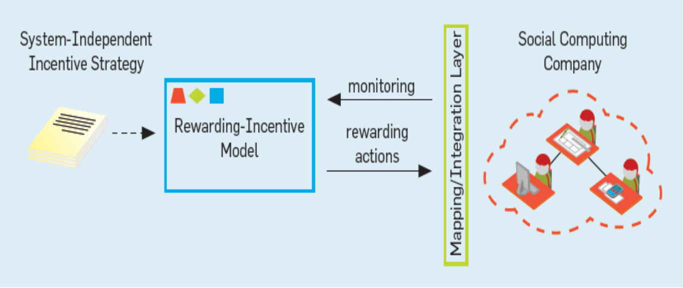
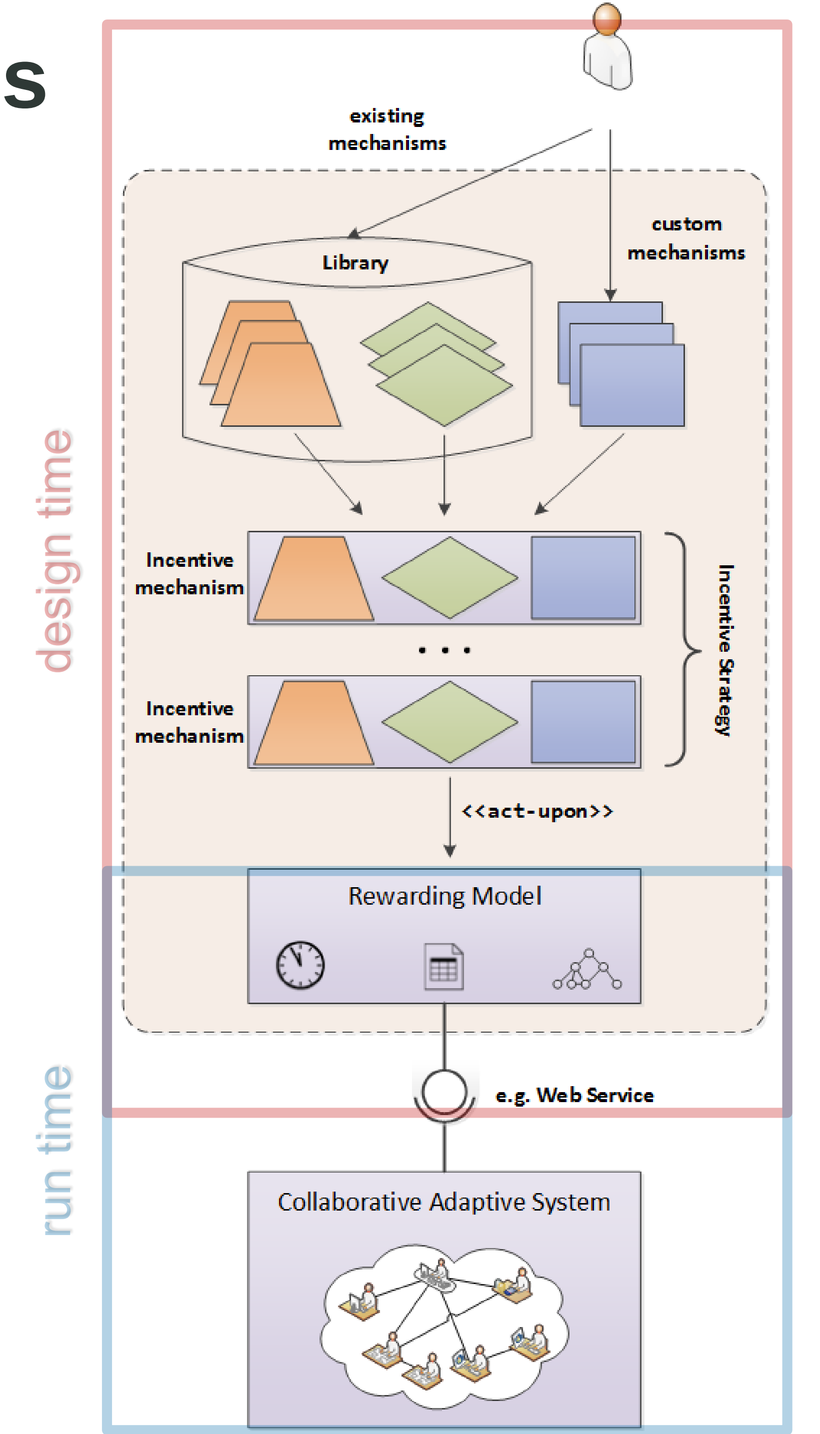


Figure 3. Conceptual scheme of a system able to translate portable incentive strategies into concrete rewarding actions for different social computing platforms.



Modeling Incentives

- We identified 7 basic incentive mechanisms in use today and their constituent elements.
- New mechanisms can be built by composing and customizing well-known incentive elements.
- Portable, reusable, scalable



Incorporate humans into a programming paradigm

Programming languages

- Abstracting human compute units as program elements
- Extending programming languages to support human compute units
- Data/control flows via extensible APIs

Multiple programming models

- Shared memory (e.g., human –software – human), message passing (human-to-human), artifact-centric, etc., via APIs working atop the compute unit abstraction layer

Execution environment

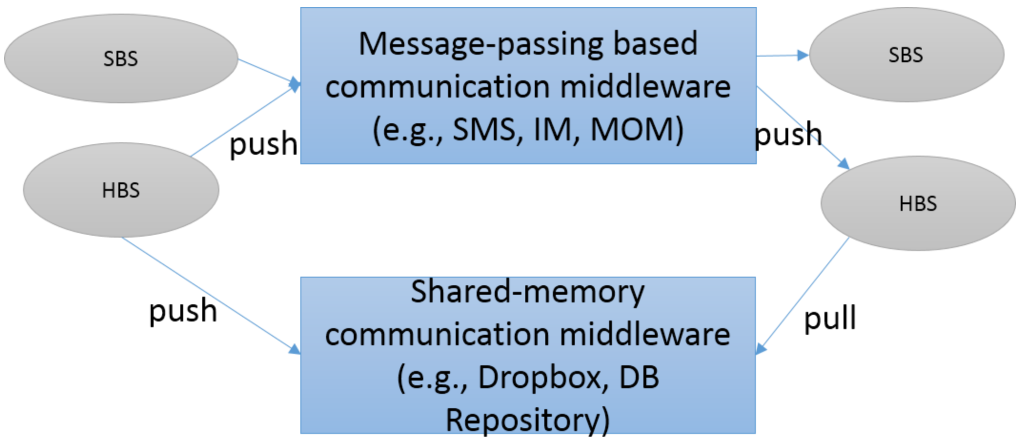
- Computing capability /profile management: human computing power, reputation and incentive models
- Monitoring and enforcing incentives/rewards, quality of results, availability
- Communication between human-middleware, among Individual Compute Units (ICU)/Social Compute Units (SCU) for exchanging artifacts and comprehensing l tasks

Cloud of hybrid service units

Cloud of HBS: A cloud of HBS includes HBS that can be provisioned, deployed, and utilized on-demand based on different pricing and incentive models.

Cloud of hybrid services: A cloud of hybrid services includes SBS and HBS that can be provisioned, deployed and utilized on-demand based on different pricing and incentive models.

HBS Communication Interface



- Based on well known technologies for integrating human requests/responses into SBS
  - Emails, web interfaces, SMS, MOM Web services, etc.
  - Can support message passing and shared memory models

Human Power Unit (HPU)

- Humans are determined via skills/skill levels
- *A particular HBS cloud* can define its own sets of skills (CS) and skill levels (SK)
  - Similar to Amazon defining its own EC unit
  - Different ways to make sure skill and skill levels declared in a cloud are consistent (e.g., via testing and monitoring).

HPU Definition: *HPU is a value describing the computing power of an HBS measured in an abstract unit. A cloud of HBS has a pre-defined basic power unit, hpu_θ, corresponding to the baseline skill bs_θ of the cloud.*

Human Power Unit (HPU) – our starting point

HPU for a particular (skill, skill level)

$$hpu(sk_i, sl_j) = hpu_\theta \times f(\frac{sk_i}{bs_\theta}) \times sl_j$$

HPU for a set of (skill, skill level), no weighted factors

$$hpu(CS(hbs)) = \sum_{i=1}^{u} hpu(sk_i, sl_i)$$

Decomposition/composition of HPU for a shared HBS

$$CS(hbs) = CS_1(hbs) \cup CS_2(hbs) \cup \cdots \cup CS_q(hbs)$$

Given an HBS, its theoretical HPU can be ***larger or smaller*** than its real HPU – unlike SBS

Human Services - Conclusions (1)

- Elastic computing is needed in different aspects
  - Scaling software, services, and people in the same application
- Not just "resource elasticity"
  - Resource, costs/benefits and quality
  - Hybrid systems of software, humans and things
- Existing quality and cost/benefits monitoring and analytics need to be extended for hybrid service units

Human Services - Conclusions (2)

- Several open research questions for realizing elastic processes of hybrid computing elements
  - Novel ***models and APIs*** are needed for integrating humans into program paradigms
  - Novel concepts for **clouds of human-based services**
  - Techniques for **combining** human-based services with software-based services
  - **Programming** elements/constructs/patterns for hybrid services
  - Hybrid service **life-cycle management**

What do we know about “elasticity”?

Elasticity in physics

"elasticity (or stretchiness) is the physical property of a material that **returns to its original shape** after the stress (e.g. external forces) that made it deform or distort is removed" – http://en.wikipedia.org/wiki/Elasticity_(physics)

- It is related to the form (the structure) of something
  - "Stress" causes the elasticity (structure deformation)
  - "Strain" measures what has been changed (amount of deformation)
- In the context of computing: given a process or a system
  - What can be used to represent "Stress" and "Strain"?
  - When does a "strain" signals a "dangerous situation"?
  - How to be elastic under dynamic "stress"?

Elasticity in economics

"elasticity is the measurement of how changing one **economic variable** affects others" – http://en.wikipedia.org/wiki/Elasticity_(economics)

- price elasticity of demand
- price elasticity of supply
- income elasticity of demand
- elasticity of substitution between factors of production
- elasticity of inter-temporal substitution

Elasticity of a function:
elasticity of y with respect to x

$$e(Y,X) = \frac{dy}{dx}\frac{X}{Y},$$

Elasticity in computing (1)

"Elastic computing is the use of computer resources which vary dynamically to meet a variable workload" –

http://en.wikipedia.org/wiki/Elastic_computing

"Clustering elasticity is the ease of adding or removing nodes from the distributed data store" –

http://en.wikipedia.org/wiki/Elasticity_(data_store)

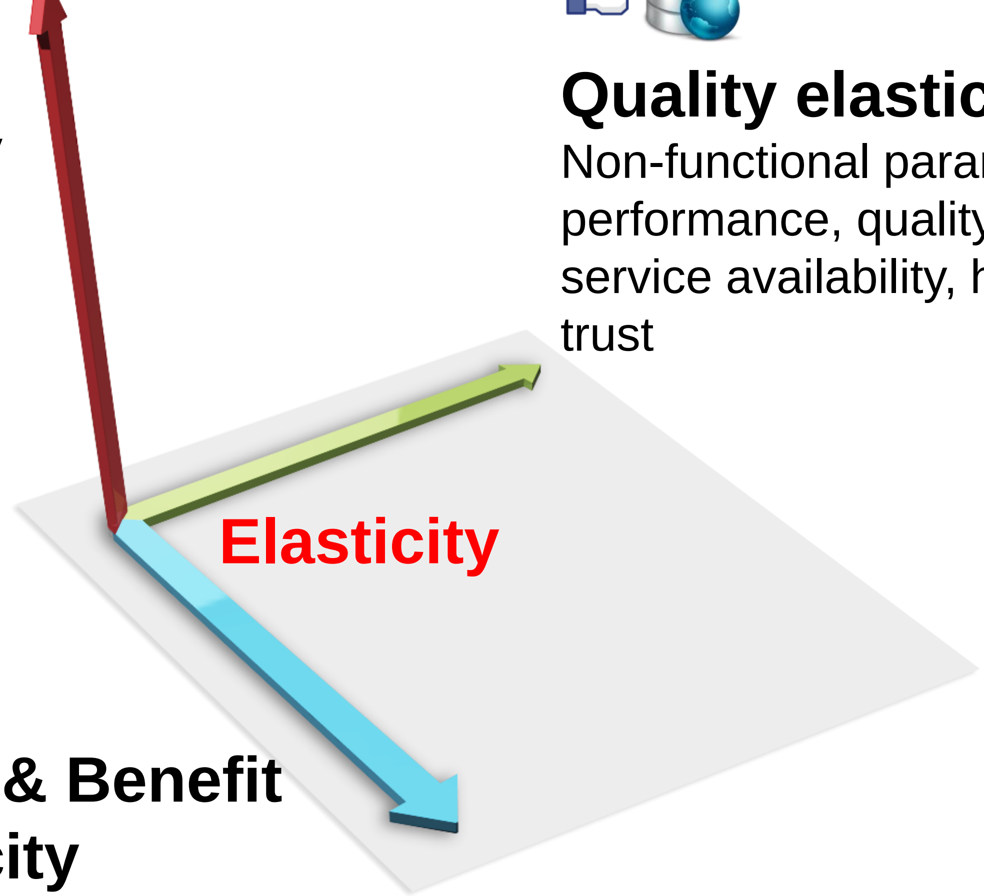
"What elasticity means to cloud users is that they should **design their applications to scale their resource requirements up and down** whenever possible.", David Chiu –

http://xrds.acm.org/article.cfm?aid=1734162

Elasticity ≠ Scaleability

Resource elasticity
Software / human-based computing elements, multiple clouds

Quality elasticity
Non-functional parameters e.g., performance, quality of data, service availability, human trust



Costs & Benefit elasticity
rewards, incentives

- **Multi-dimensional Elasticity**
- **Service computing models**
- **Cloud provisioning models**

Schahram Dustdar, Hong Linh Truong: Virtualizing Software and Humans for Elastic Processes in Multiple Clouds- a Service Management Perspective. IJNGC 3(2) (2012)

Elasticity in computing – broad view

1. Demand elasticity

   Elastic demands from consumers

2. Output elasticity

   Multiple outputs with different price and quality

3. Input elasticity

   Elastic data inputs, e.g., deal with opportunistic data

4. Elastic pricing and quality models associated resources

Diverse types of elasticity requirements

- ***Application user***: "If the cost is greater than 800 Euro, there should be a scale-in action for keeping costs in acceptable limits"
- ***Software provider***: "Response time should be less than amount X varying with the number of users."
- ***Developer***: "The result from the data analytics algorithm must reach a certain data accuracy under a cost constraint. I don't care about how many resources should be used for executing this code."
- ***Cloud provider***: "When availability is higher than 99% for a period of time, and the cost is the same as for availability 80%, the cost should increase with 10%."

Vienna Elastic Computing Model

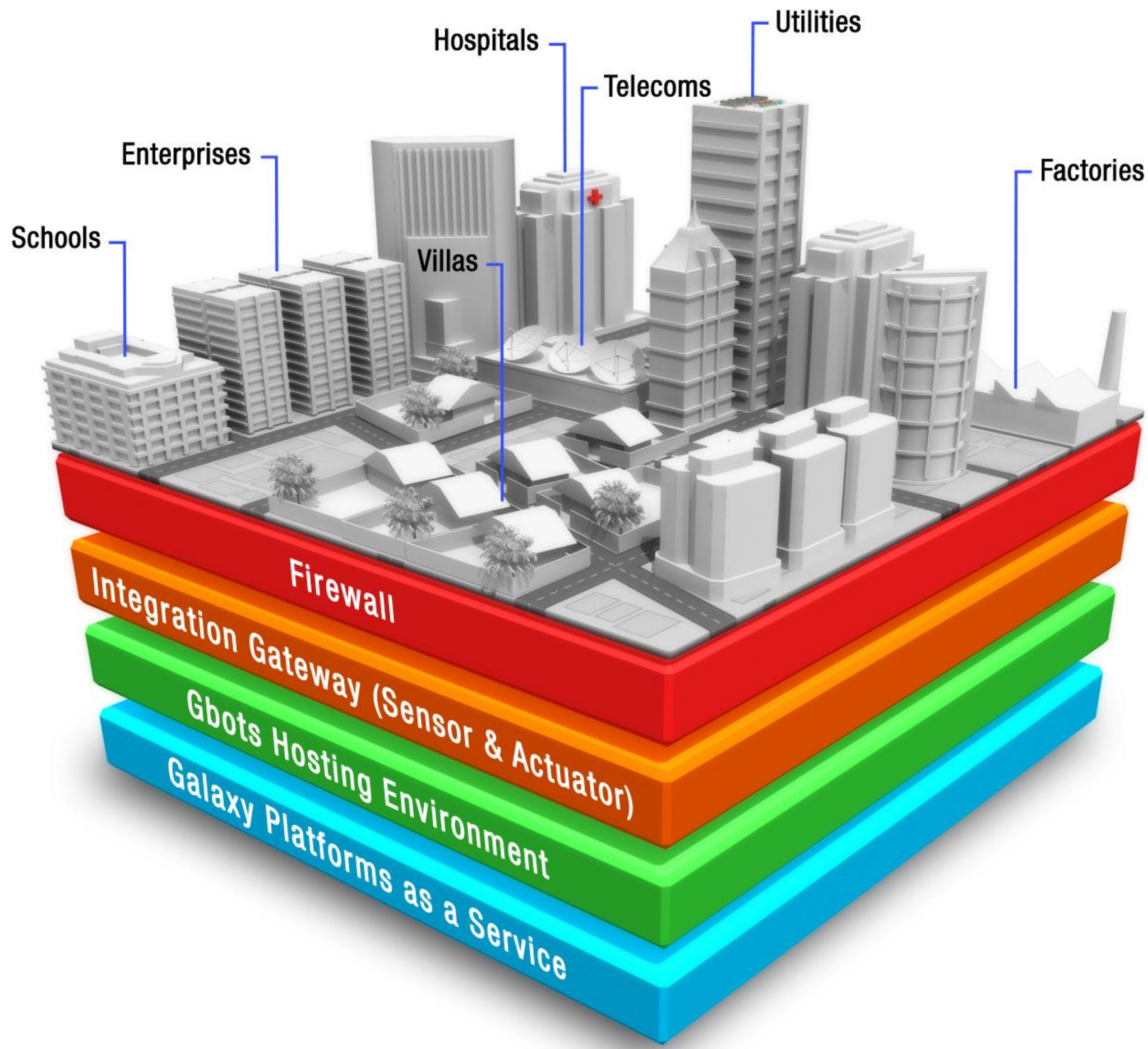
dsg.tuwien.ac.at/research/viecom

- Multi-dimensional elasticity
  - Resource, quality, cost and rights
    - Please see "Principles of elastic processes" paper
- Elasticity in hybrid systems of human-based and software-based computing resources
  - Software and human capabilities as computing resources
  - Multi clouds
- End-to-end approach
  - For science and business complex applications

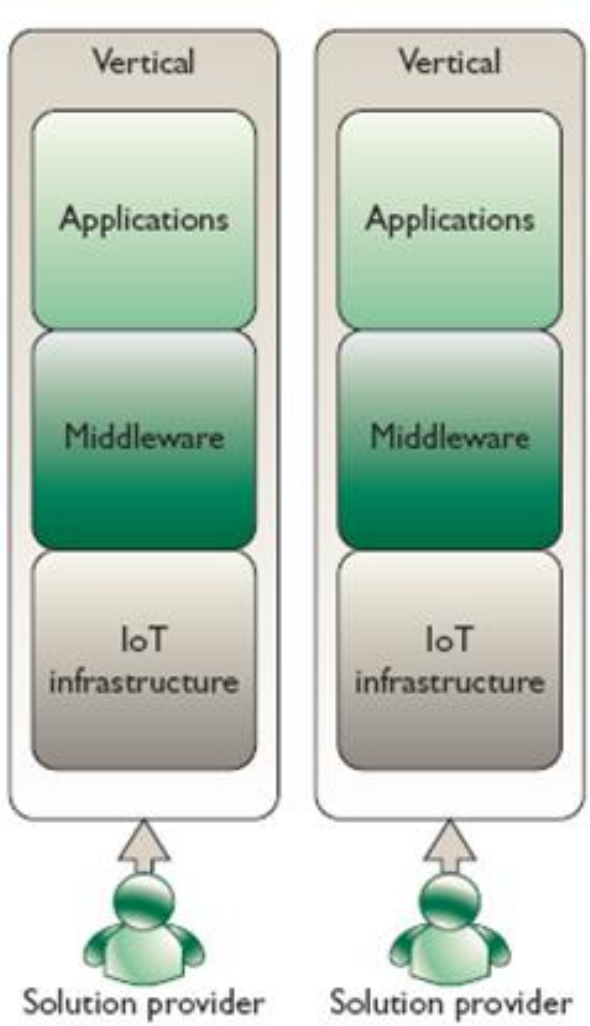
Things

as a service

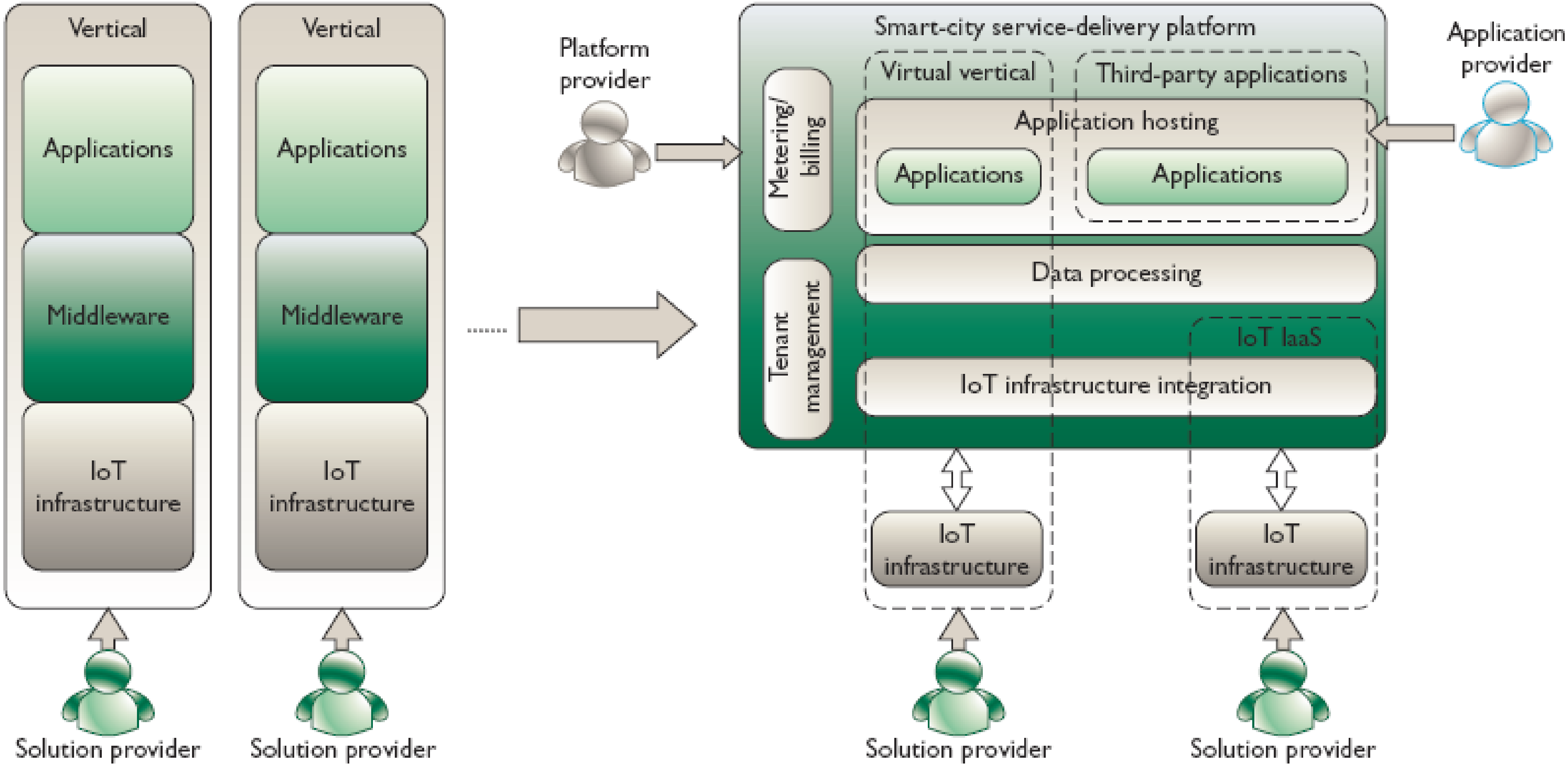
Managed City Governance Service Oriented Architecture


Utilities
Hospitals
Telecoms
Enterprises
Factories
Schools
Villas
Firewall
Integration Gateway (Sensor & Actuator)
Gbots Hosting Environment
Galaxy Platforms as a Service

Vertical IoT solutions today



Smart Vertical solutions tomorrow?



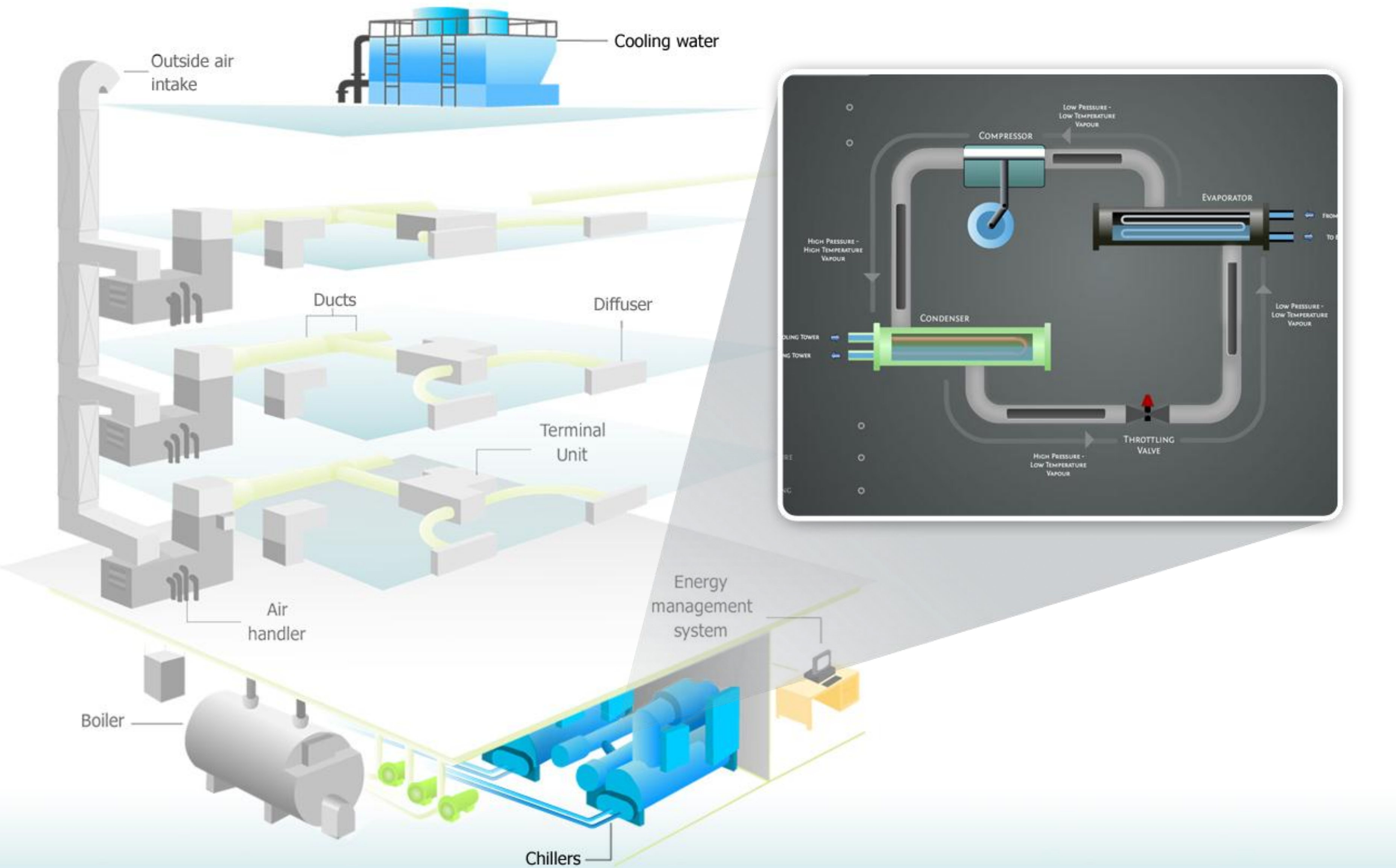

Ministry of Interior
Directorate General of Civil Defence
24x7
DUBAI CIVIL DEFENCE
24x7 Direct Alarm System

Emergency Hub

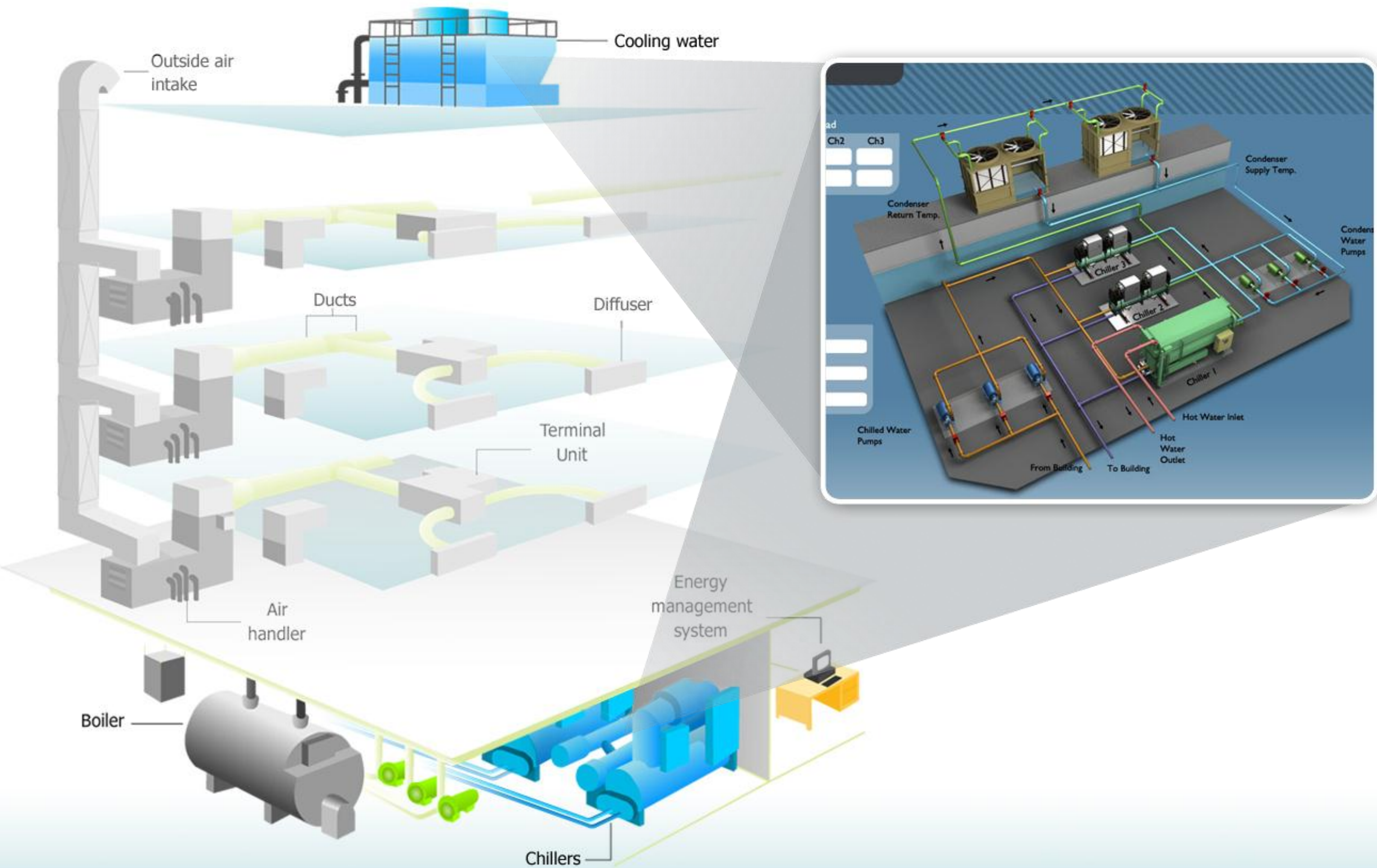
ICT for energy savings in buildings

Command Control Center for Managed Services

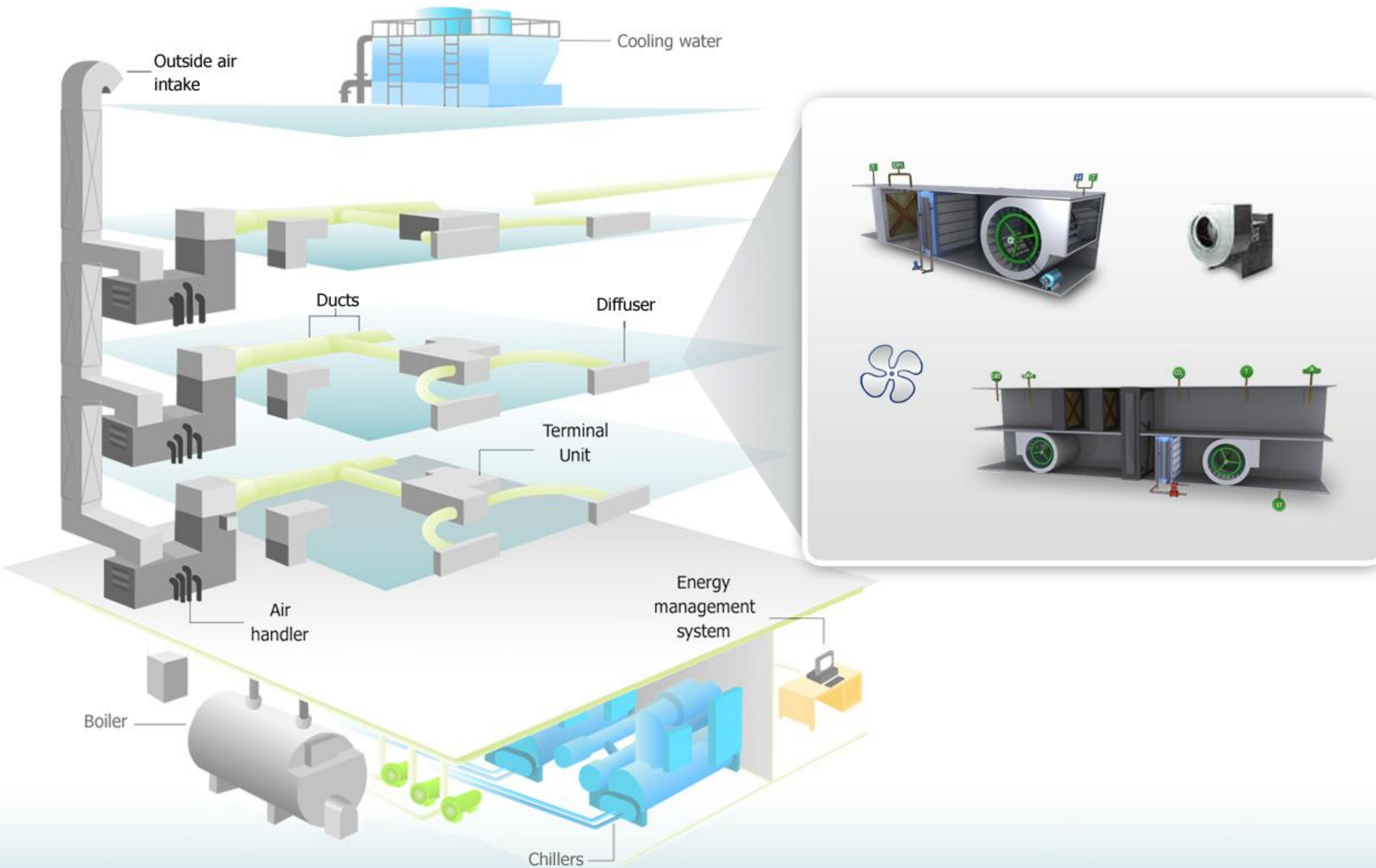
HVAC (Heating, Ventilation, Air Conditioning) Ecosystem



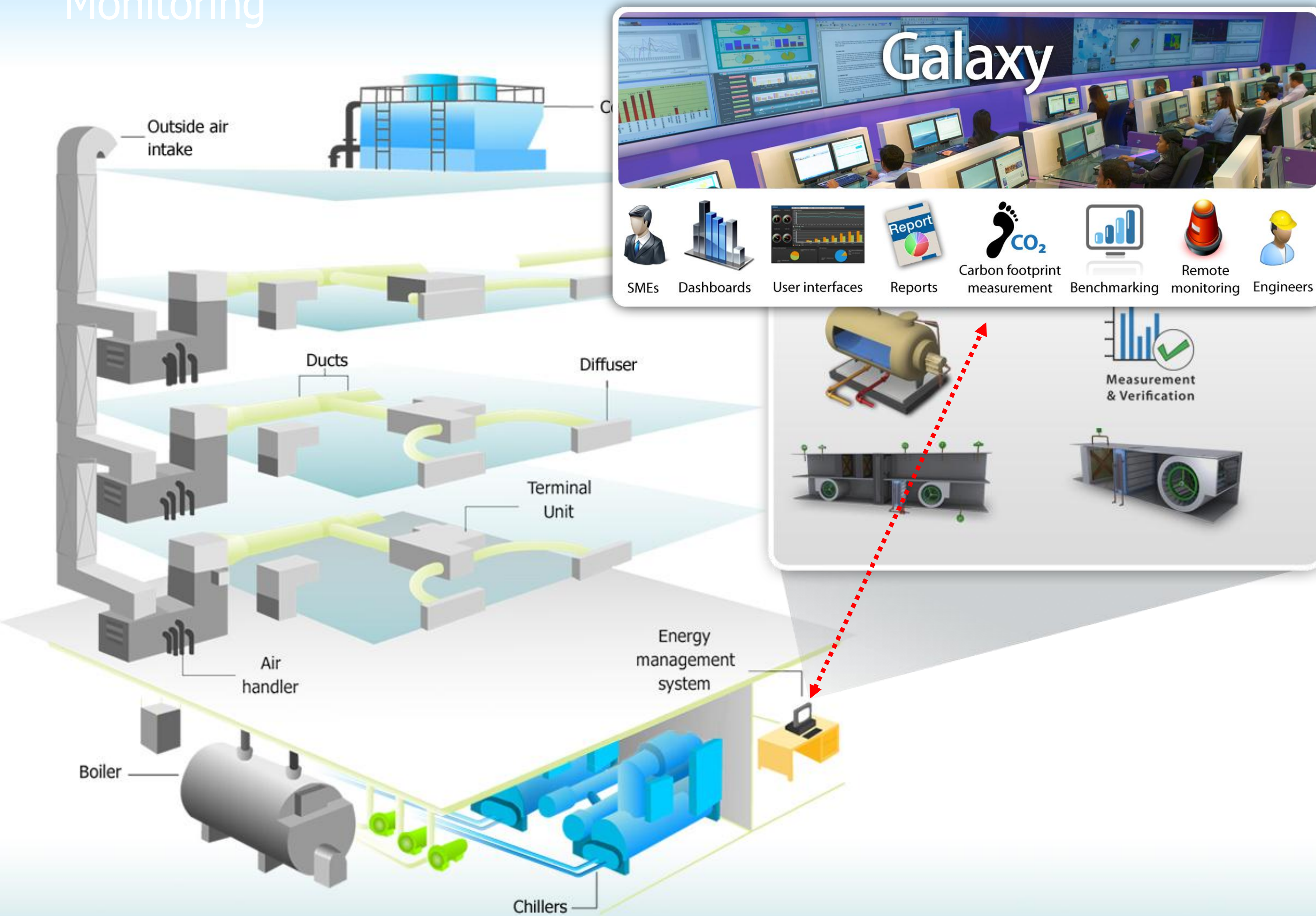
Water Ecosystem



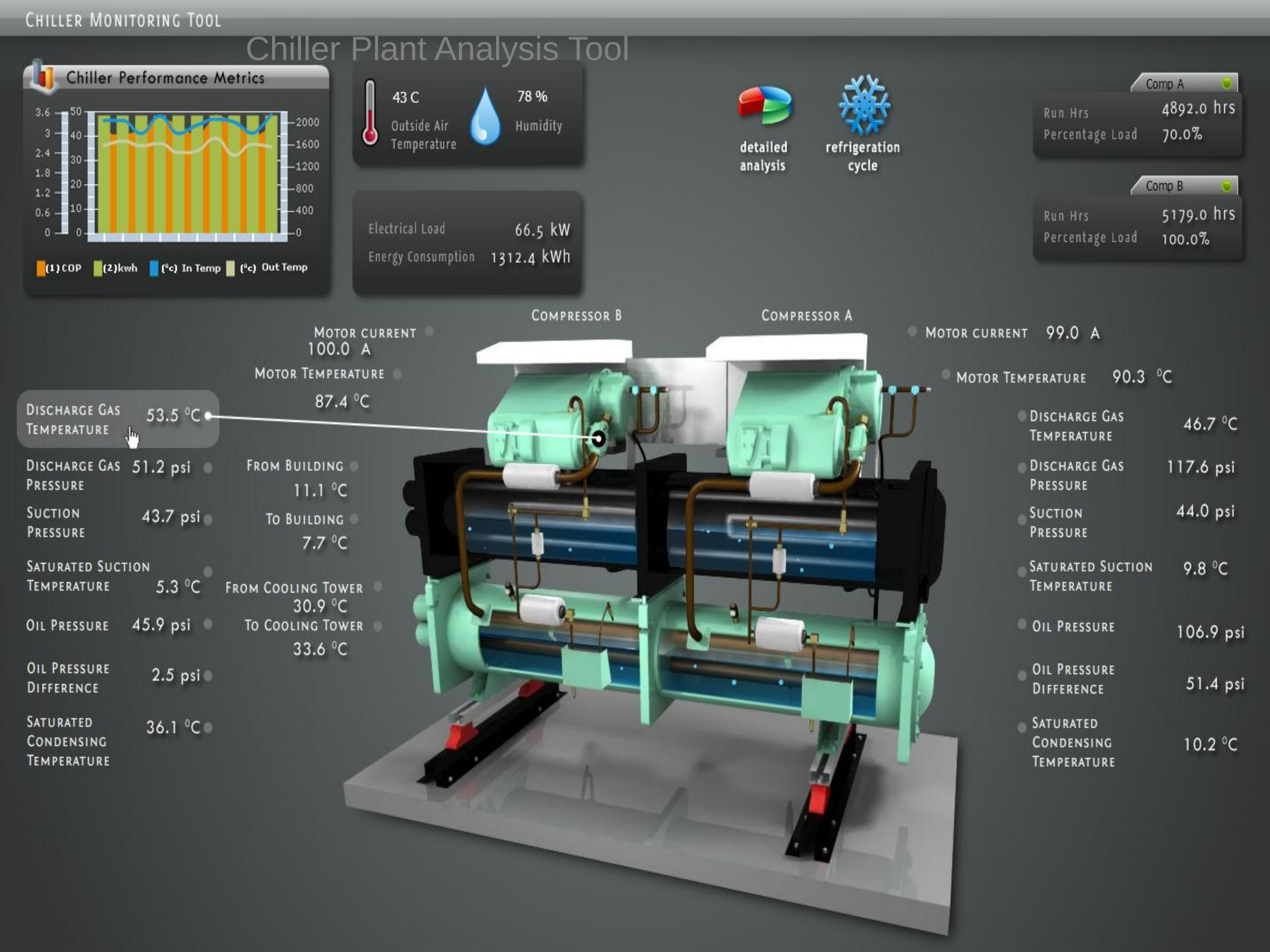
Air Ecosystem

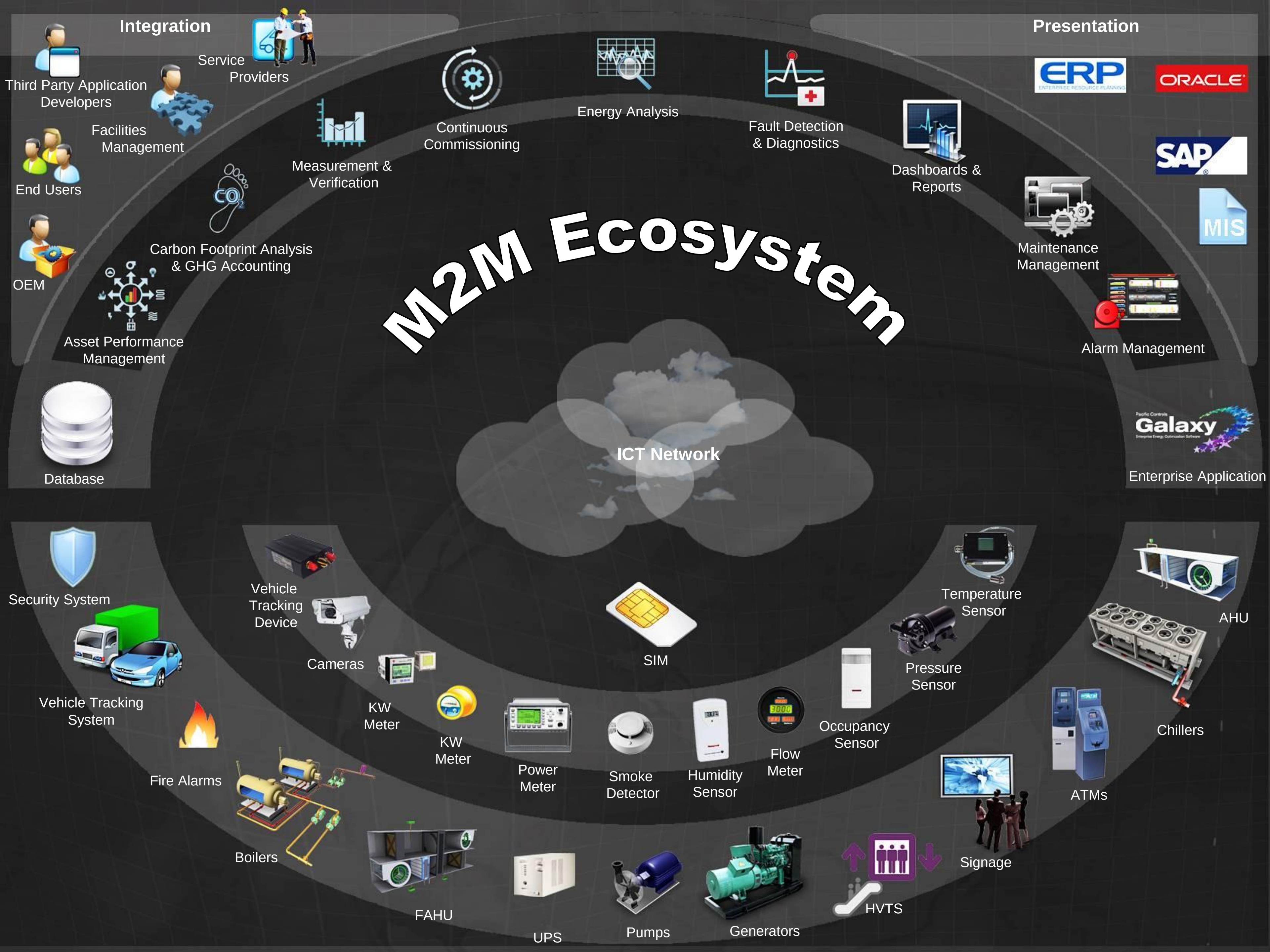


Monitoring


Outside air intake
Ducts
Diffuser
Terminal Unit
Air handler
Energy management system
Boiler
Chillers
Galaxy
SMEs
Dashboards
User interfaces
Report
Reports
CO2
Carbon footprint measurement
Benchmarking
Remote monitoring
Engineers
Measurement & Verification

Chiller Plant Analysis Tool




M2M Ecosystem
Integration
Presentation
Third Party Application Developers
Service Providers
Facilities Management
End Users
OEM
Carbon Footprint Analysis & GHG Accounting
Measurement & Verification
Continuous Commissioning
Energy Analysis
Fault Detection & Diagnostics
Dashboards & Reports
Maintenance Management
Alarm Management
Asset Performance Management
ERP
ORACLE
SAP
MIS
Database
ICT Network
Galaxy
Enterprise Application
Security System
Vehicle Tracking System
Fire Alarms
Boilers
FAHU
UPS
Pumps
Generators
HVTS
Signage
ATMs
Chillers
AHU
Vehicle Tracking Device
Cameras
KW Meter
KW Meter
Power Meter
Smoke Detector
Humidity Sensor
Flow Meter
Occupancy Sensor
Pressure Sensor
Temperature Sensor
SIM

Remote Service Maintainence

Elasticity in smart environments

1. Elastic human demands: utilizing human-based services with Social Compute Units
2. Output elasticity: using multi-scale analytics + data-as-a-service as elastic resources
3. Elastic data inputs for urban mobility and energy systems: using data-as-a-service as elastic data resources
4. Elastic pricing and quality models: utilizing bots and cloud analytics

Thanks for your attention!

Prof. Dr. Schahram Dustdar

Distributed Systems Group
TU Wien

dsg.tuwien.ac.at