Architecting digital repeatable systems for systemic Digital Transformation

Module 1-1

DT of large-scale digital repeatable systems

Dr Alexander Samarin

SAMARIN.BIZ

There is a strong demand for sustainable future

- The Club of Rome
- Sustainable Development Goals from the UN
- Everything is requested being "smart"
  (or sustainable – can meet the needs as long as wanted)
- Essential requirements for cities
  - "Cities" refers to **any geographically located population** (village, town, megapolis, island, valley, etc.)
  - combining **diversity** and **uniformity**
  - **be scalable**, e.g. 4 500+ cities (with 150 000+ citizens) nee become smart in a sustainable way
  - employ **digital**

- It is a **systemic problem** to be solved in a **systemic manner (IEC uses "Systems Committee" approach)**

International, regional and national standardisation

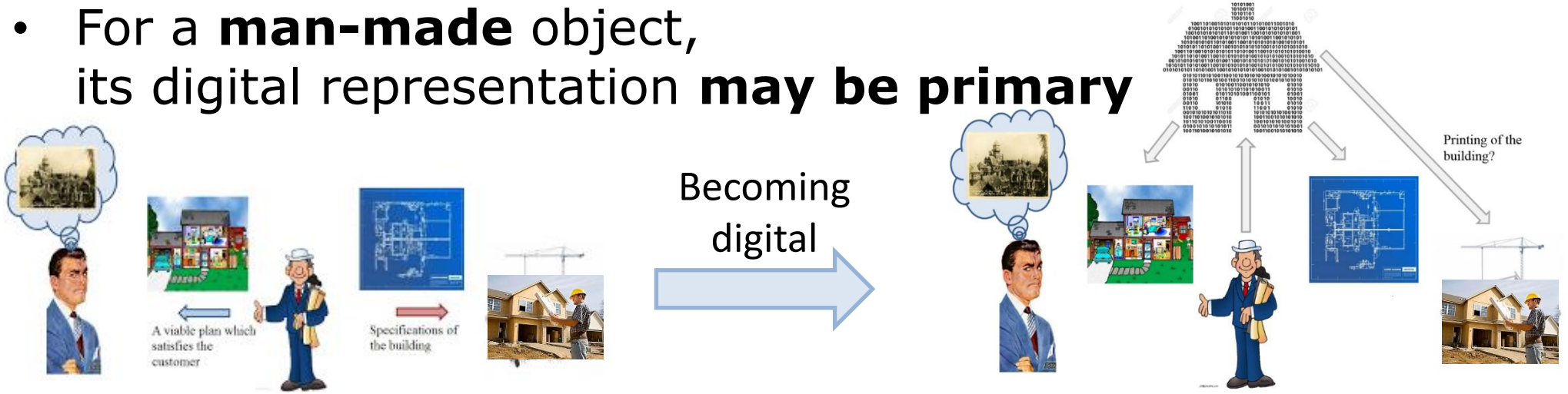
- ISO – International Organisation for Standardisation
- IEC – International Electrotechnical Commission
- ITU – International Telecommunication Union
- ISO/IEC JTC 1 – Joint Technical Committee on IT
- CEN/CENELEC – European analogue of ISO and IEC
- IEEE – The Institute of Electrical and Electronics Engineers
- ANSI – The American National Standards Institute
- BSI – The British Standards Institution
- AFNOR – Association Française de Normalisation
- DIN – Deutsches Institut für Normung
- БИС – Български институт за стандартизация

Systems work at IEC

- IEC has established “Systems Committees” (in addition to “Technical Committees”) consider some domains systemically
  - SyC Smart Energy
  - SyC AAL – Active Assisted Living for people with disabilities and elderly
  - SyC Smart Cities
  - SyC LVDC – Low Voltage Direct Current
  - SEG 8 – Communication Technologies and Architectures of Electrotechnical Systems
  - SEG 9 – Smart Home/Office Building Systems
  - SEG 10 – Ethics in Autonomous and Artificial Intelligence Applications
  - SEG 11 – Future Sustainable Transportation

What is digital (representation)

- An object may have **many representations** at the same time
  - physical, analogue, digital, bionic, etc.
- **Digital representation** is explicit, formal, computer-readable and computer-executable
- For a **nature-made** object,
  its digital representation **is secondary**
- For a **man-made** object,
  its digital representation **may be primary**



Many representations of a house

House design (digital) v1

Model

Improve

House design (digital) v2

Built house (digital)

Implement

Monitor

Improve

Built house (physical) v1

Control Alerts

Built house (physical) v2

Time

Current understanding of digital: it is always “secondary” never “primary”

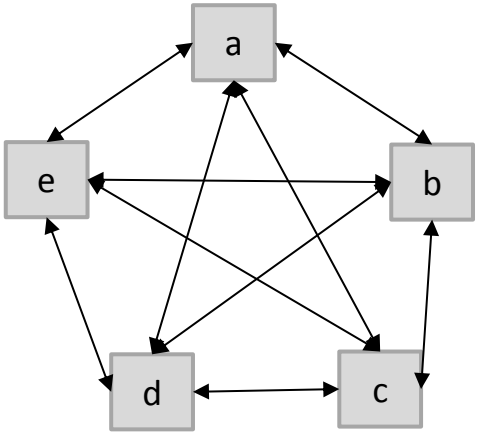
- “Digital twin”, “Digital footprint” or “Digital shadow”
  - a digital twin is a digital replica of a living or non-living physical entity
- Thus Digital Transformation (DT) cannot bring all its benefits
- Let us consider the digital representation as primary one
  - Intelligent Transportation System
  - Software defined communication network
  - Software defined network + wireless sensors network
  - Modern autonomous car
  - Software defined enterprise https://bpm.com/blogs/executable-architecture-of-software-defined-enterprises

From software-defined systems to digital systems

- Any modern system is a system of various types of interwoven and interdependent "sub-systems" (similar to body anatomy)
  - social systems
  - economic systems
  - technical systems
  - biological systems
  - intellectual systems
  - physical systems
  - ethic systems
  - cyber-physical systems
  - real-time systems
  - software-intensive systems
  - information systems
  - classic computing systems
- These sub-systems **must be aligned** – via their digital representations (primary or secondary)
- Let us consider "**digitally coordinated systems**" (or digital systems) – systems which are architected, governed, managed, operated on the basis of digital representation of its elements, features and relationships between them

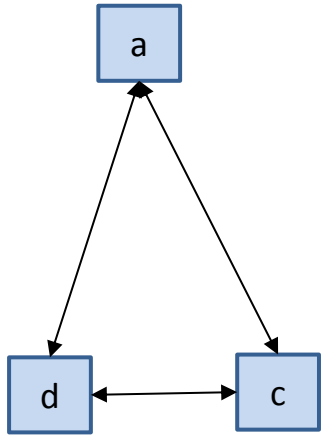
The main secret of digital - easy to create necessary variations (1)

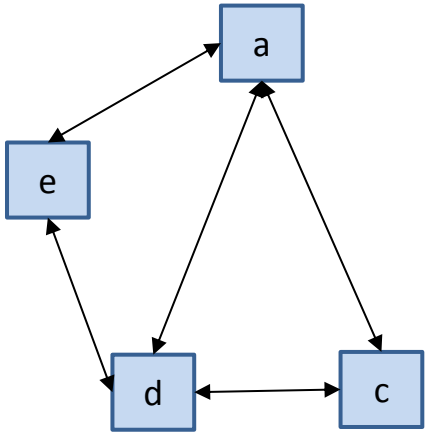
1. A digital element is easy repeatable (cloneable); **copy cost is (almost) zero**
2. A universal (with rich functionality) digital system which satisfies many customers is **very difficult to create and evolve** (“monolith” negative pattern)
3. A digital system is a coherent set of digital elements connected in a digital way, since any **system consists of elements and connections between them** (“assembly” pattern)



The main secret of digital - easy to create necessary variations (2)

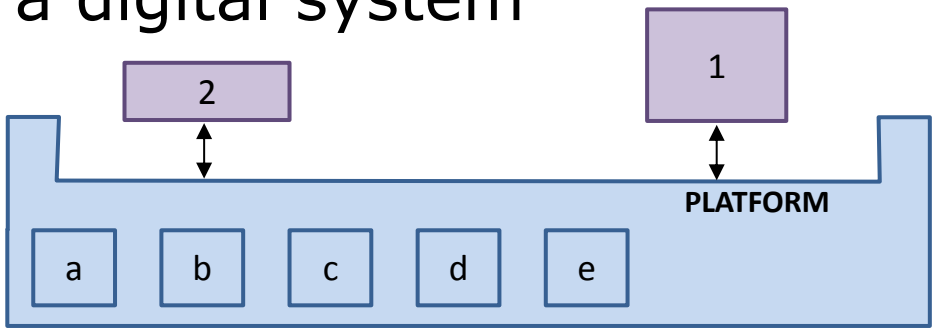
4. Individual versions of digital systems can be easily **assembled from standard digital repeatable elements** (“Lego” pattern)



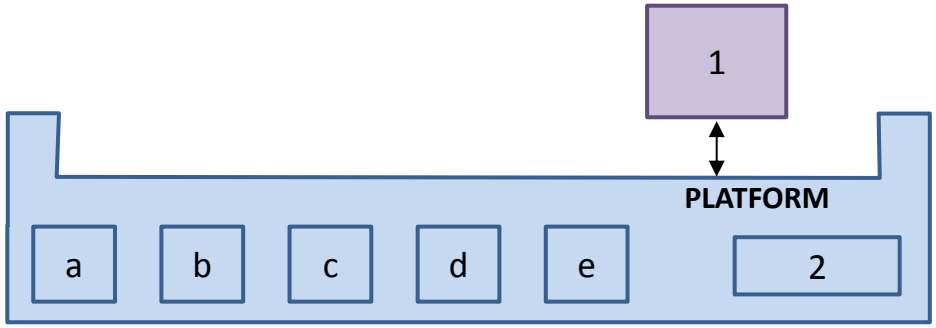


The main secret of digital - easy to create necessary variations (3)

5. Custom digital elements (labelled “1” and “2”) can be **quickly added as needed** thus allowing the creation of all sorts of individual versions of a digital system (“platform” pattern)

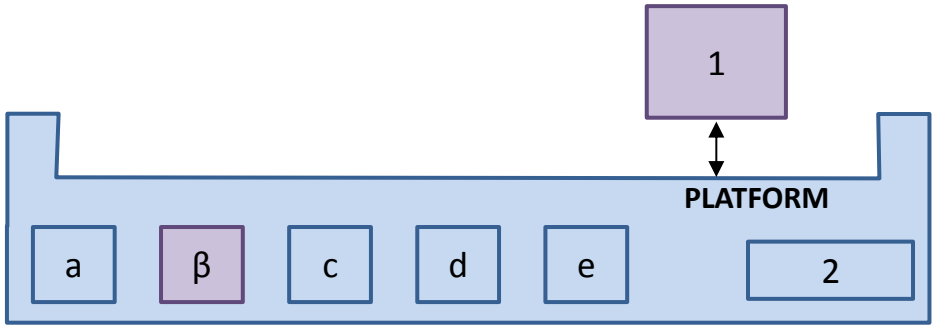


6. If some custom digital elements **become popular** then they can be included in the platform for wider distribution (“software factory” pattern)



The main secret of digital - easy to create necessary variations (4)

7. When necessary a platform component **can be replaced** by another component (labelled "β") which follows the same interfaces ("API" pattern)



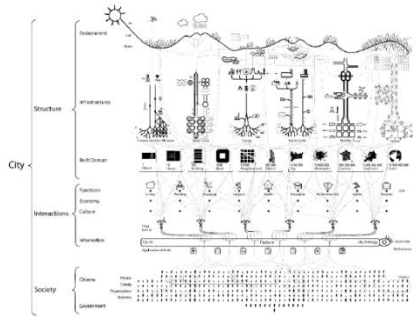
- Thus, the most efficient and effective way of DT is
  - coordinate and complement creating digital repeatable elements
  - use methods for assembling digital repeatable systems from digital repeatable elements – standard (common) and custom (unique)
  - reuse (sell, rent, copy) such elements and systems many times

How to transform a system to the digital one

- Systemically
  - coherent viewpoints, views, descriptive and **executable models**
- Value-driven
  - understand pains and cures
  - understand **flows of value**, their performance and potentials
- Via Human Enterprise Learning Leadership (HELL)
  - **everyone is an important stakeholder** who must learn and act
- Trustworthiness by design
  - privacy **by design**, quality by design, etc.
- With coordination, complementation and copying
  - to address next uber-complex challenges with limited talent resources

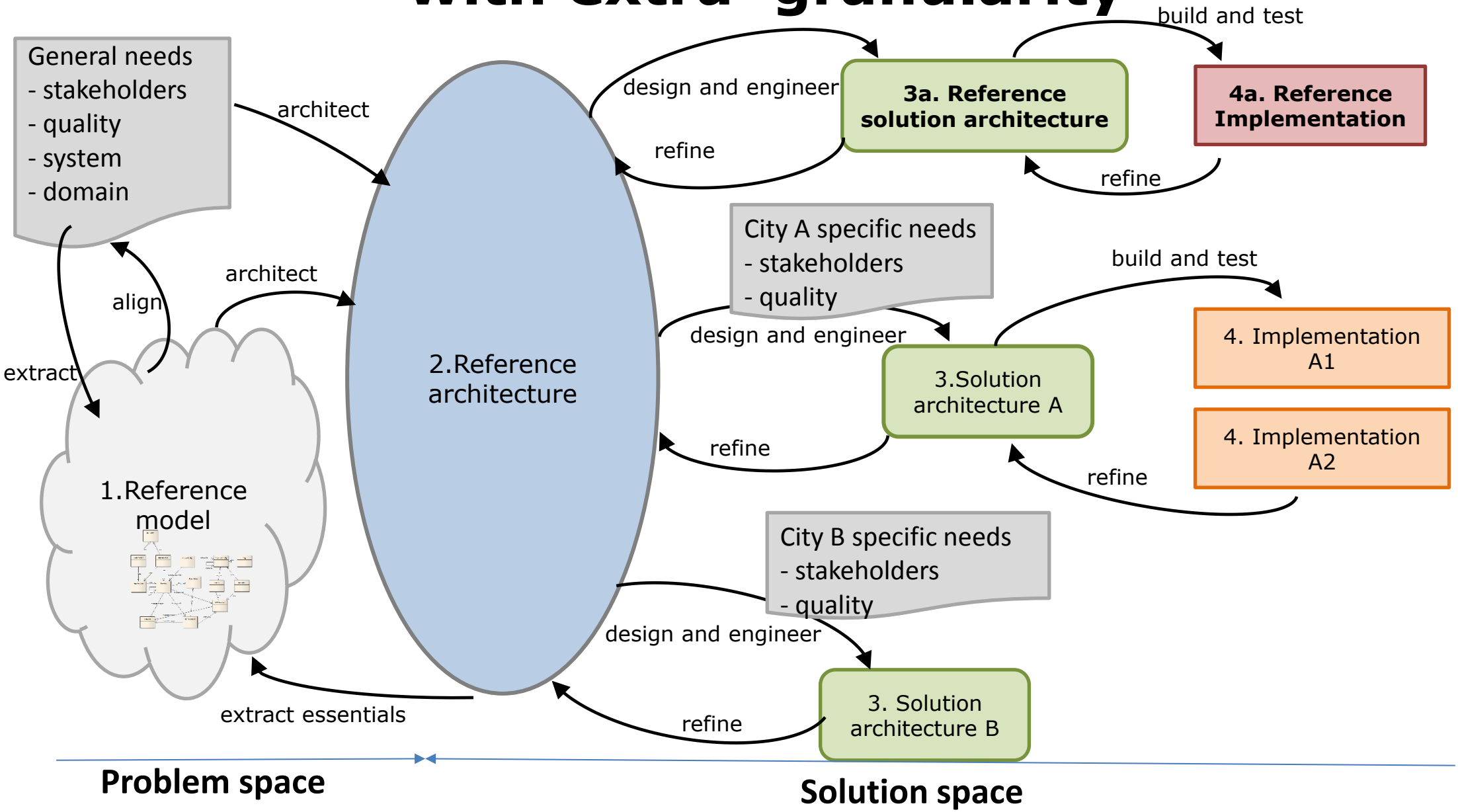
Cities complexity

- Unpredictable and unlimited growth and development
- Each city is different; all cities have some commonalities
- Digital data and information in huge volumes
- Contradictory demands for security and privacy
- Many diverse stakeholders
- Distributed and decentralised
- Great influence on our society
- Multidisciplinary; many flows; many rules



- City is a system of various interwoven systems (social, economic, technical, physical, intellectual, biological, etc.)
- **Smart city is a city (re)built as a digitally coordinated repeatable system**

Pattern "Levels of architecting" with extra granularity



Pattern “Reference Architecture”: a common need for a sustainable solution

Citizens
Society
Business
Government

Citizens
Society
Business
Government

Citizens
Society
Business
Government

Let us

1) Build common understanding
2) Isolate common parts
3) Find how to integrate unique and common parts
4) Develop common parts once and with high quality as a platform
5) Have a version of the common platform at each Smart City
6) Cooperate, complement and copy among Smart Cities

Together Smart Cities will gain a lot in quality, time and money

Architecting DT for Smart Cities: a common understanding of common parts

Citizens
Society
Business
Government

A | A

Citizens
Society
Business
Government

B | B

Citizens
Society
Business
Government

T | T

Industries

Best practices

Academic and research institutes

Best knowledge

IEC/SyC **ISO/TC** **JTC1** **ISO/PC** **IEC/TC** **…**

Smart Cities Reference Architecture (SCRA) and Smart Cities Reference Architecture Methodology (SCRAM)

Best standards

Standards Development Organizations (IEC, ISO, ITU, JTC1, IEEE, …)

IEC SyC Smart Cities WG3

Common parts

Unique parts

SCRAM & SCRA
1) all digital solutions are repeatable
2) each city may add its own solutions under the common methodology, i.e. SCRAM

Architecting DT for Smart Cities: repeatable solutions for common parts

Citizens
Society
Business
Government

Citizens
Society
Business
Government

Citizens
Society
Business
Government

S1 | A 2 | S3
City A
digital repeatable platform

B1 | S2 | B3
City B
digital repeatable platform

T1 | T3
City T
digital repeatable platform

Primary market:
City digital transformation

Primary market:
City digital transformation

Telecommunication providers

IT companies (start-ups, local, global)

Coordination, complementating and copying of digital repeatable solutions and systems

Secondary market:
Platform implementation

Secondary market:
Platform implementation

S1 | **S2** | ... | **S3**
Common Smart Cities digital repeatable platform

Industries

Financial organisations

Academic and research institutes

IEC/TC | IEC/SyC | ISO/TC | JTC1 | ISO/PC | ...
SCRA and SCRAM

Standards Development Organizations (IEC, ISO, ITU, JTC1, IEEE, …)

IEC SyC Smart Cities WG3

Common parts

Unique parts

Architecting DT for Smart Cities: investment opportunities

Citizens
Society
Business
Government

Citizens
Society
Business
Government

Citizens
Society
Business
Government

S1 | A 2 | S3
City A
digital repeatable platform

B1 | S2 | B3
City B
digital repeatable platform

T1 | T3
City T
digital repeatable platform

Estimations are provided (mid 2019) by company Relex (www.relex.io) which is specialised in real estate crowdfunding

Coordination, complementating and copying of digital repeatable solutions and systems

S1 | S2 | ... | S3
Common Smart Cities
digital repeatable platform

IEC/TC | IEC/SyC | ISO/TC | JTC1 | ISO/PC | ...
SCRA and SCRAM

Smart Cities FUND

10% yield

Ordinary investors and crowdfunding

Institutional investors

Platforms acts an amplifier for 4 500+ cities

First comers advantage

Eliminate negative yield of several trillion USD market

City infrastructure insurance companies

Common parts

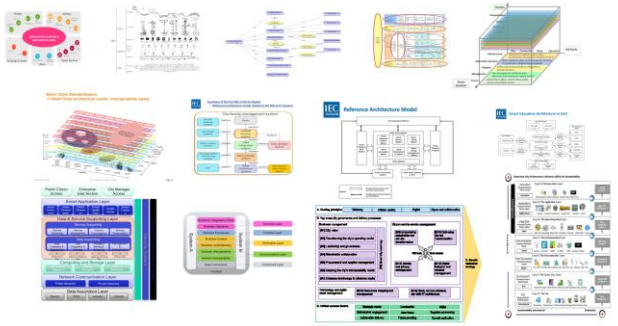
Unique parts

Simple calculations

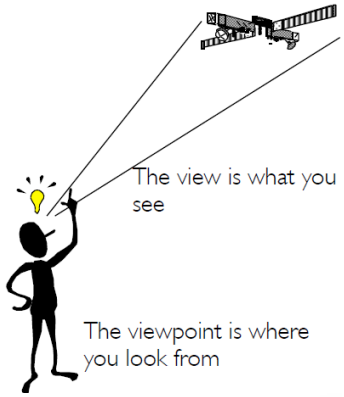
- **N** is the total cost of a Smart City implementation (construction)
- **70 %** - common, **30 %** - unique
- Total cost for 100 Smart Cities **WITHOUT standardisation**
  - **N * 100**
- Total cost for 100 Smart Cities **WITH standardisation**
  - N * 100 * 0.3 (unique parts) +
    N * 1 * 0.7 (common parts) * 3 (complexity factor) =
    N * (30 + 2.1) =
    N * 32.1
- Cost difference is (N*100) / (N*32.1) ≈ **3 times**!
- Maintenance and evolution will be much cheaper as well

How to describe such architecture?

- **Many disciplines** to be used together
- The whole system **life cycle**
  - conception, development, production, utilization, support, retirement and destruction
- There is **no one single framework** which covers all of these phases
- However, there are many frameworks which are "**monolith**" (also known as "silos")
  - ZF, TOGAF, PEAF, POET, FEAF, DoDAF, MoDAF, NAF, RM-ODP, JTC1/SC7 software engineering standards, CoBIT, ITSM (ITIL), ISO 20000, ISO 27000, ISO 9000, BIZBok, BABok, BPMBoK, PMBok and many other "disciplines" such as DevOps, Agile, SCRUM, etc.
- Smart Cities Reference Architecture Methodology (**SCRAM**)
- Smart Cities Reference Architecture (**SCRA**)

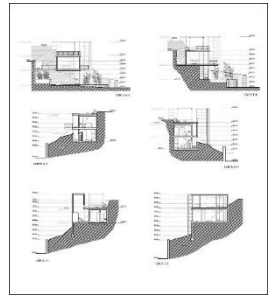


ISO/IEC/IEEE 42010:2011 architecture description



Views (system-in-focus dependent) are governed by **viewpoints** (system-in-focus independent)

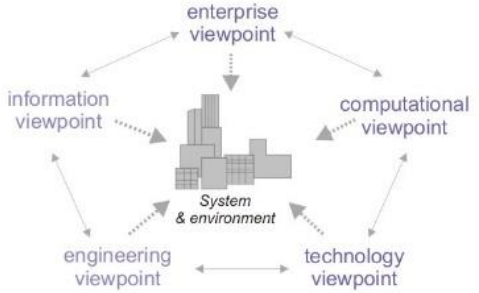
Geometrical views of buildings are viewed side by side — as a **composition**



Architecture views are often originated by different people — thus they **must be aligned** to be used together

Each view comprises one or many **models**. Any model consists of **artefacts** (e.g. applications, servers, products, reports, etc.) and relationships between them.

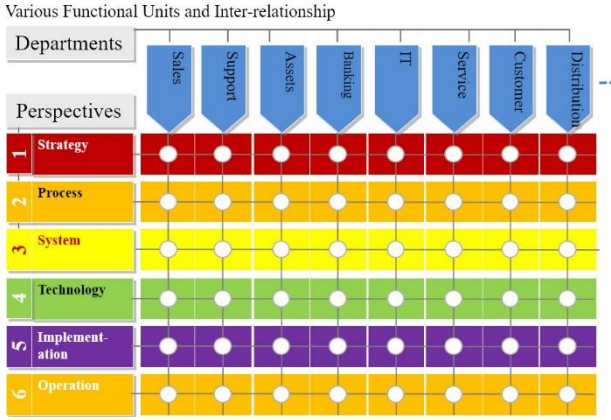
Models (system-in-focus dependent) are governed by **model-kinds** (system-in-focus independent).

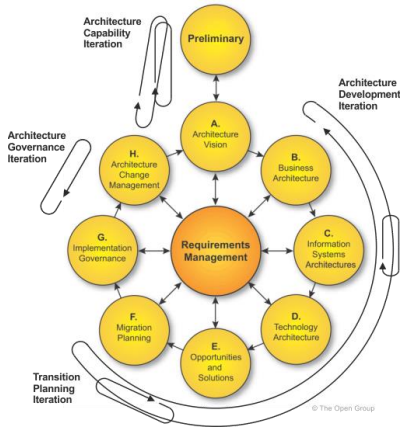


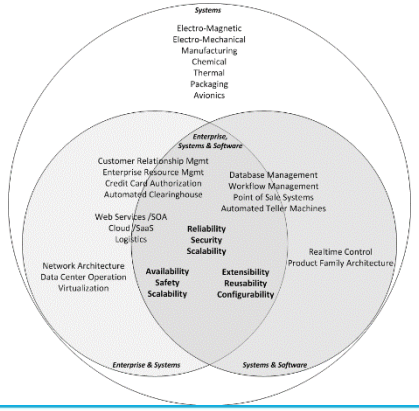
SCRAM: Collection of viewpoints, model-types, artefacts-types and patterns

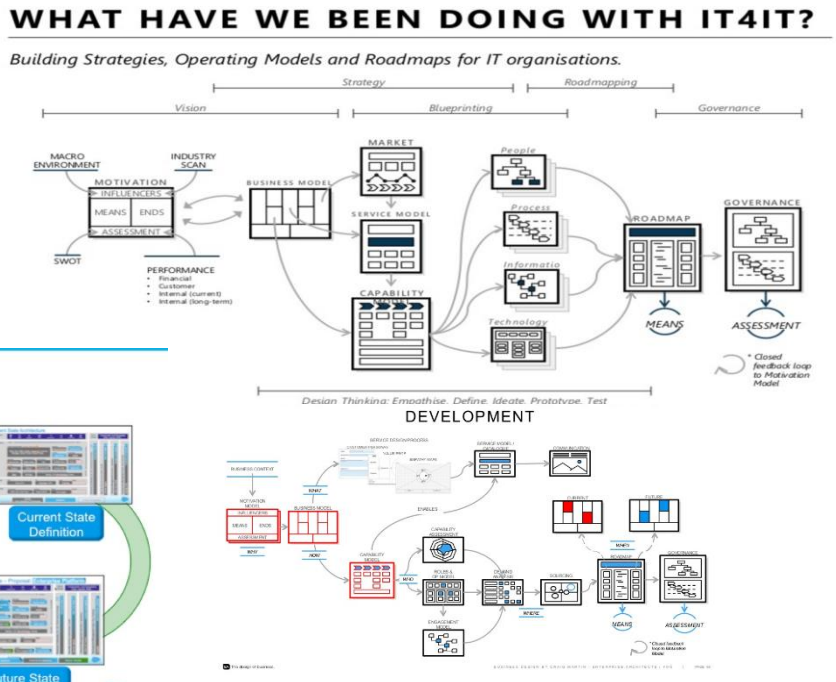
- We created the SCRAM the following way
  - Decomposed many "monolith" frameworks into smaller pieces
  - Sorted those pieces out
  - Structured those pieces
- **SCRAM viewpoints** collect one or many SCRAM model-types
- **SCRAM model-types** link one or many SCRAM model-types and/or **SCRAM artefact-types**
- **SCRAM patterns** are methods to create an SCRAM model-type from other SCRAM model-types
- If possible **models are digital**, i.e. formal, explicit, machine-readable and machine-executable

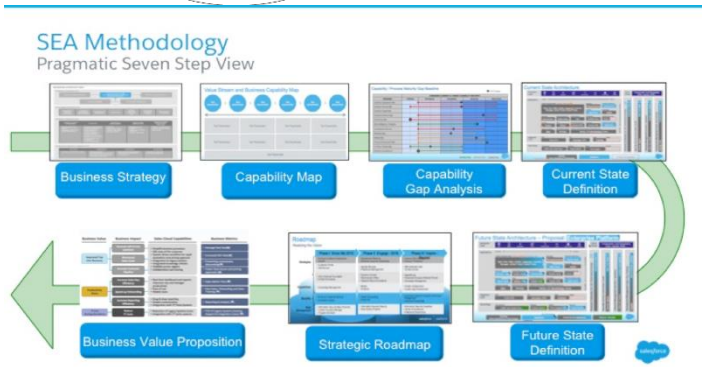
All frameworks comprise many model-kinds

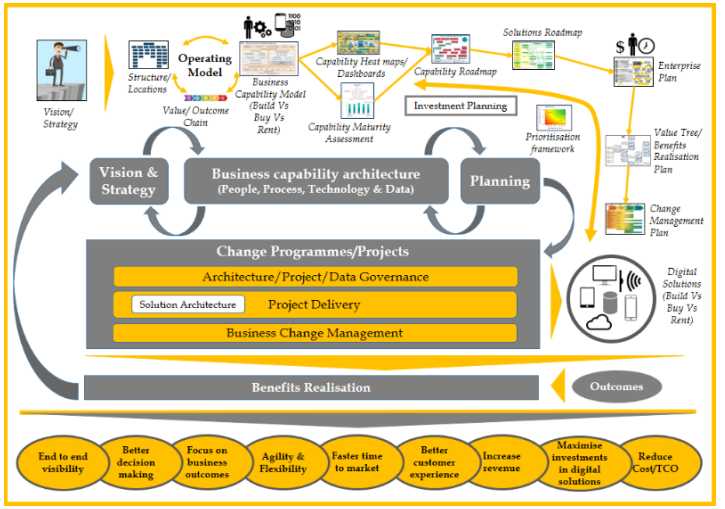






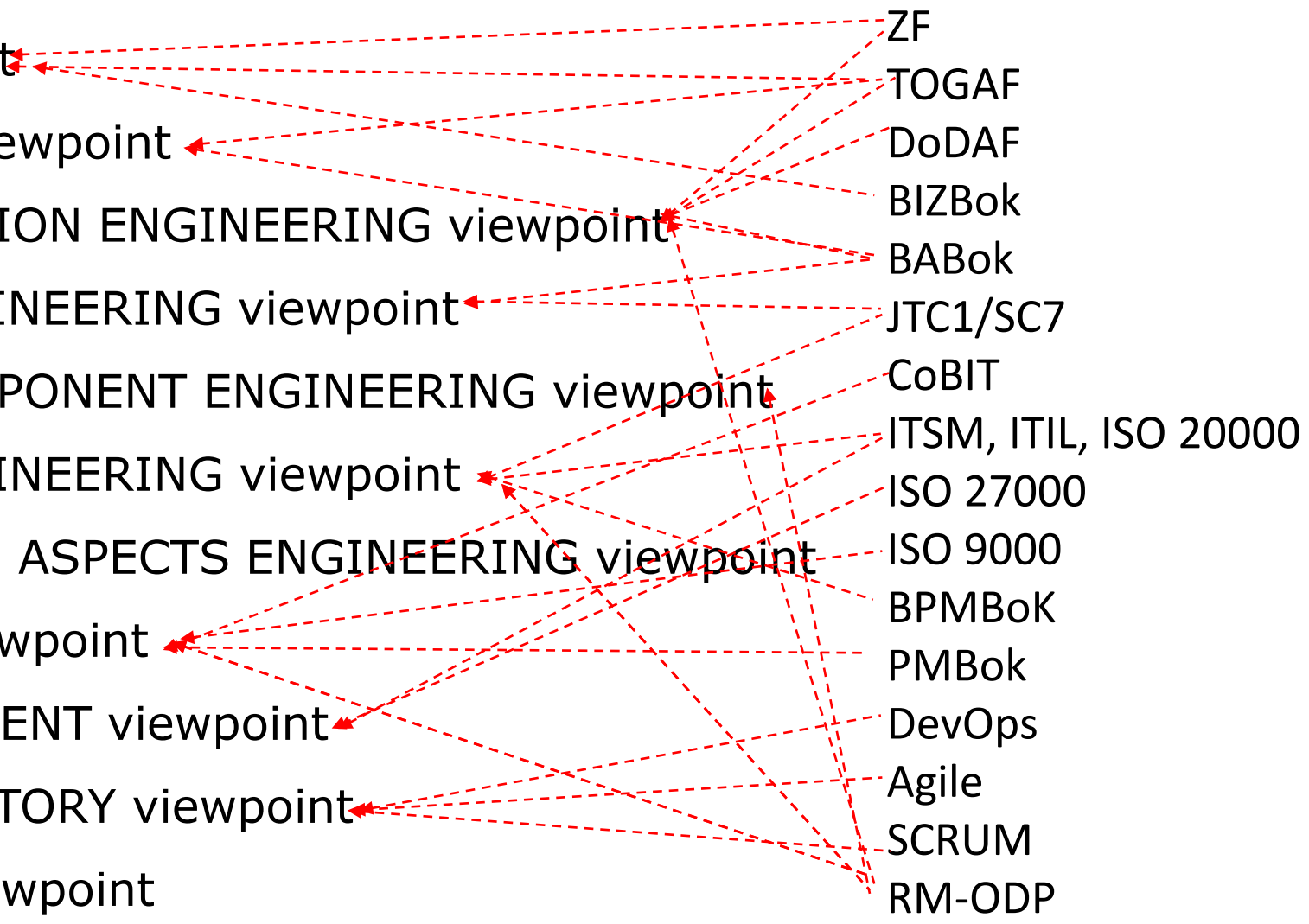




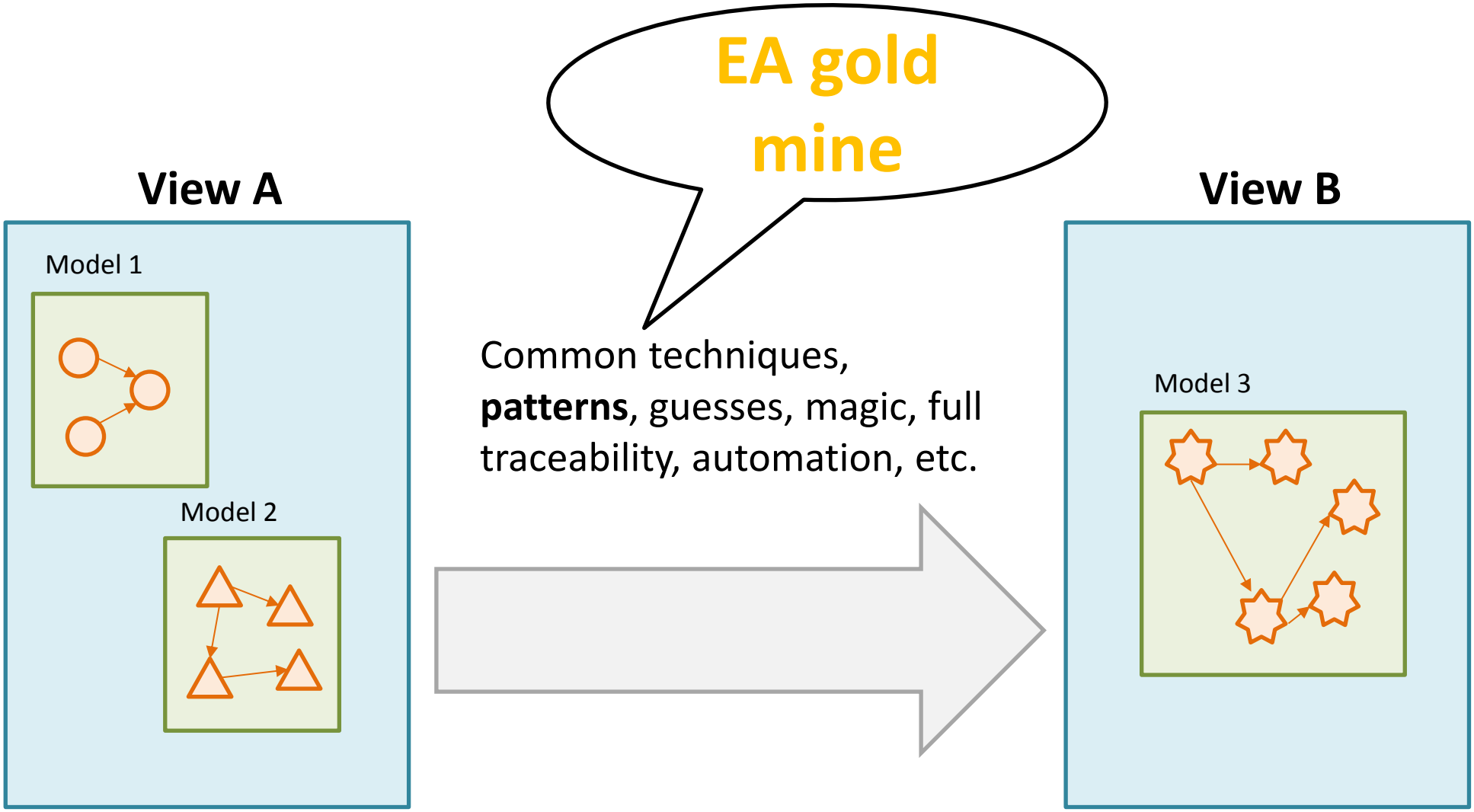


SCRAM: a set of viewpoints (11) and model-types (107)

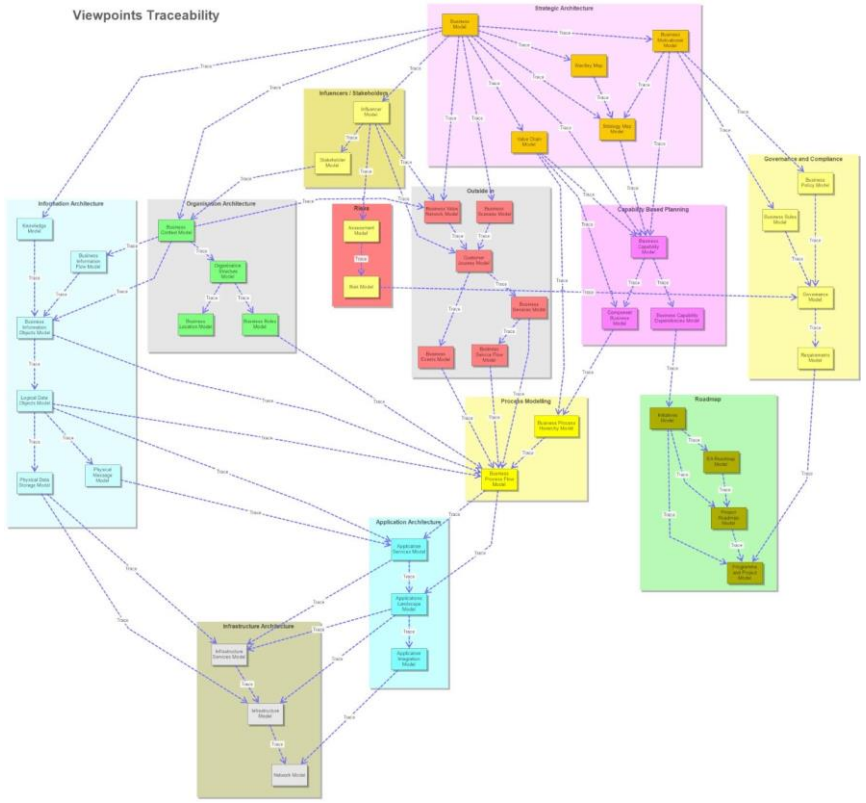
- VALUE viewpoint
- BIG PICTURE viewpoint
- SYSTEM-SOLUTION ENGINEERING viewpoint
- PLATFORM ENGINEERING viewpoint
- PLATFORM COMPONENT ENGINEERING viewpoint
- SOLUTION ENGINEERING viewpoint
- CROSSCUTTING ASPECTS ENGINEERING viewpoint
- CORPORATE viewpoint
- RISK MANAGEMENT viewpoint
- SOFTWARE FACTORY viewpoint
- STANDARDS viewpoint



SCRAM/SCRA: Some models may be generated from others



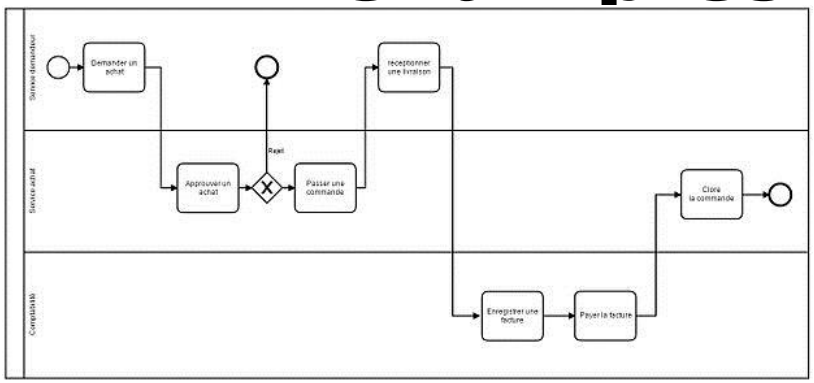
SCRAM: Dependencies between viewpoints and model-types



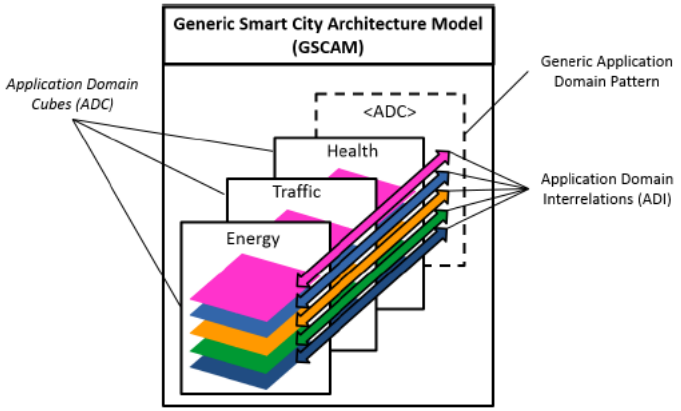
- There is a “happy path”
- Actually, it will be “pinball” because if something has been changed that all connected elements must to be validated

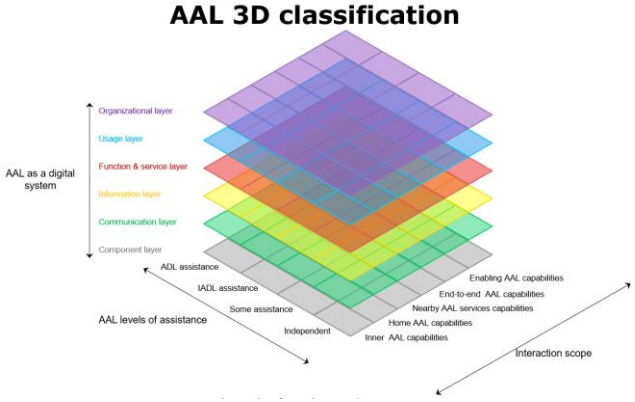
SCRAM: Complex dependencies examples

- Process

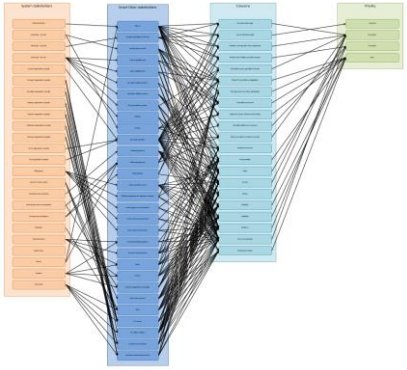


- Classifications

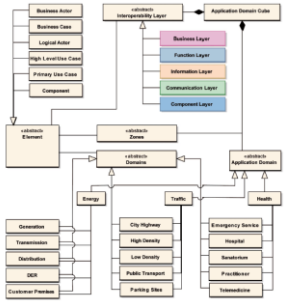




- Assembles



- Tensors



- General schemas

SCRAM vs SCRA

SCRAM is a set of architecting rules

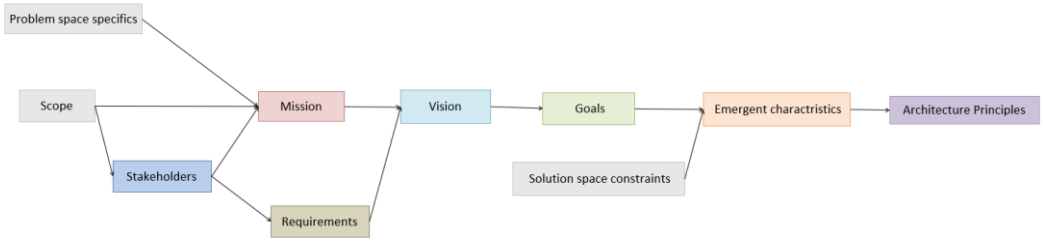
SCRA is an architecture of an idealized city

SCRAM viewpoints —govern→ SCRA views

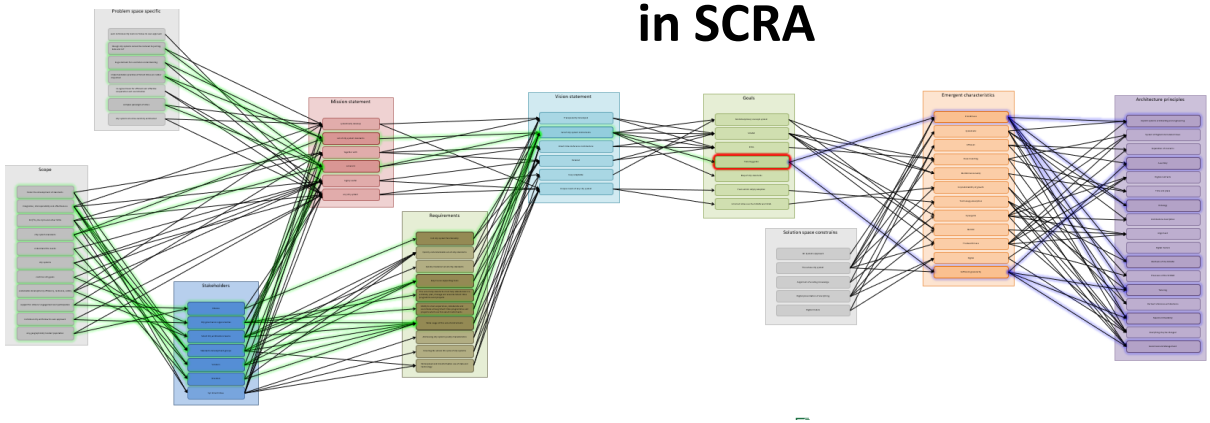
SCRAM model-types —govern→ SCRA models

SCRAM artifact-types —govern→ SCRA artifacts

in SCRAM



in SCRA



SCRA: VALUE view guiding principles (example)

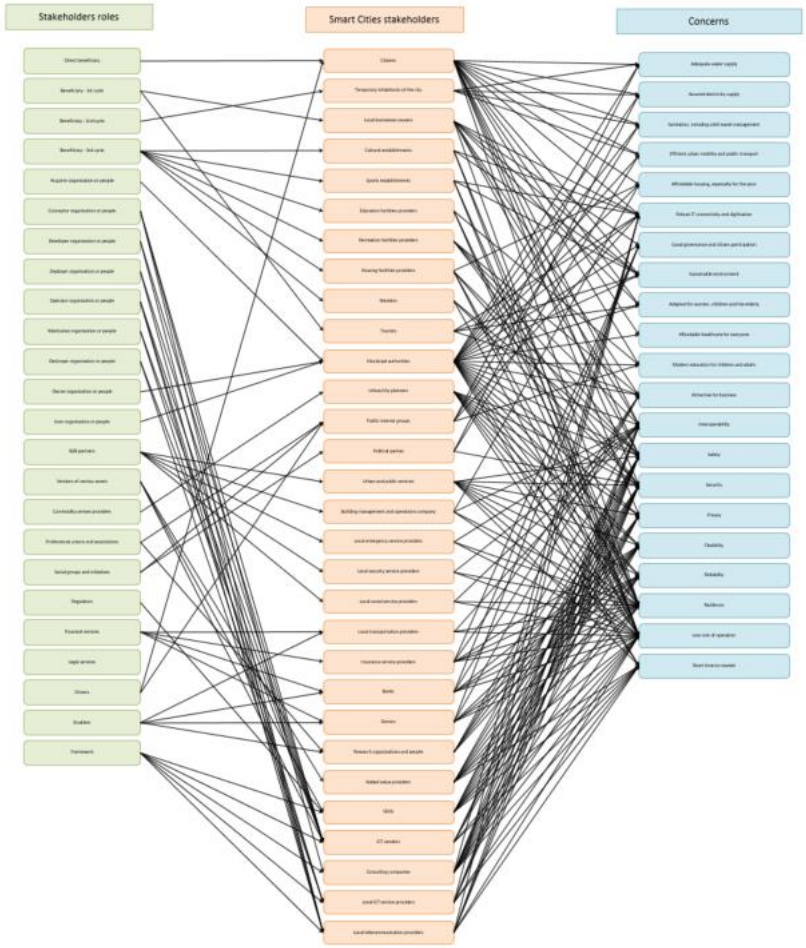
- The guiding principles for defining Smart Cities architectures are
  - interoperability
  - safety
  - security (including confidentiality, integrity and availability)
  - privacy
  - resilience
  - simplicity
  - low cost of operation
  - short time to market
  - combining diversity and uniformity
  - self-referential

SCRA: Value view high-level requirements (example)

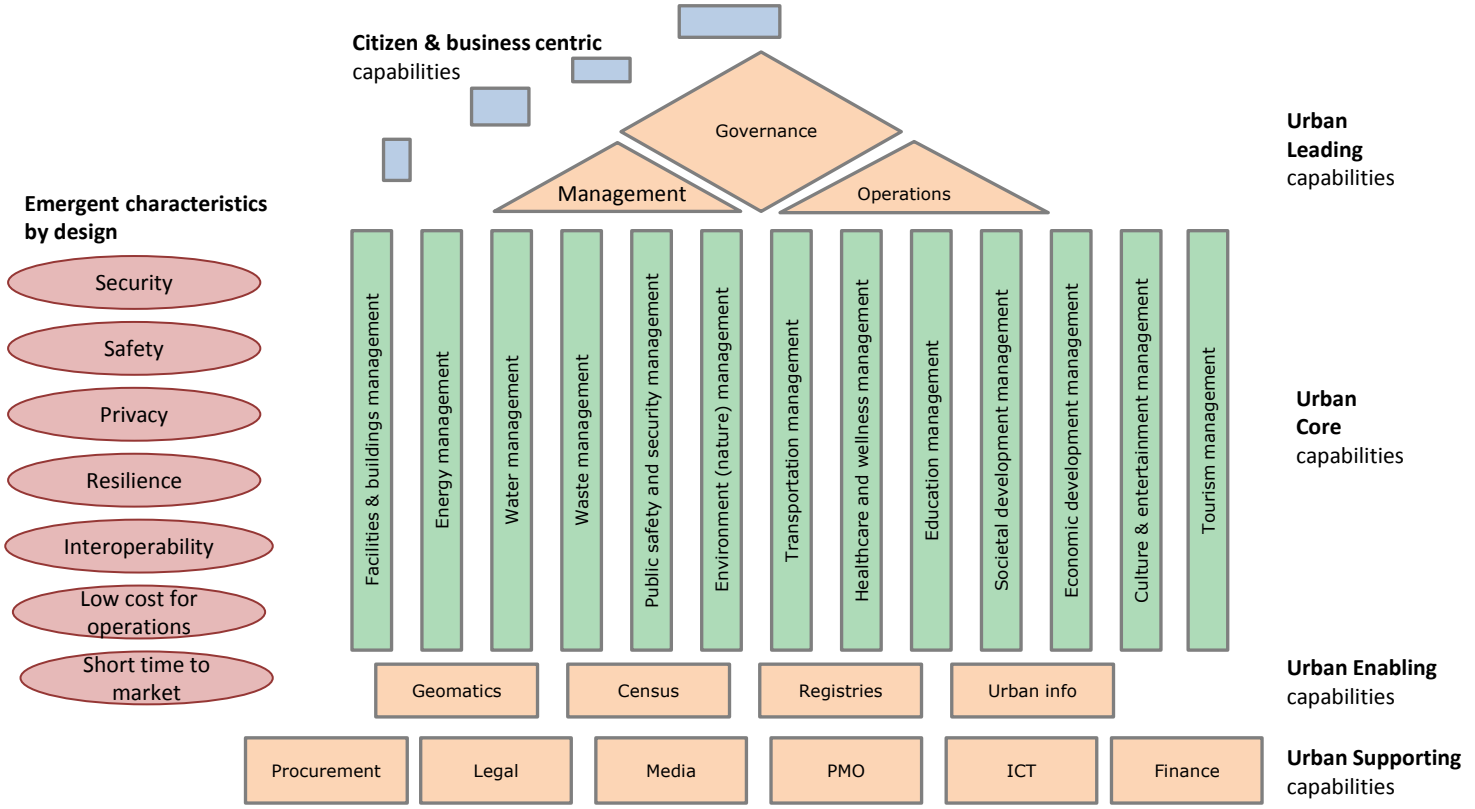
- List of high-level requirements
  - Adequate water supply
  - Assured electricity supply
  - Sanitation, including solid waste management
  - Efficient urban mobility and public transport
  - Affordable housing, including for the poor
  - Robust IT connectivity and digitalisation
  - Good governance and citizen participation
  - Sustainable environment
  - Safety and security of citizens, particularly women, children and the elderly
  - Affordable healthcare for everyone
  - Modern education for children and adults
  - Attractive for business

SCRA: stakeholders' concerns analysis

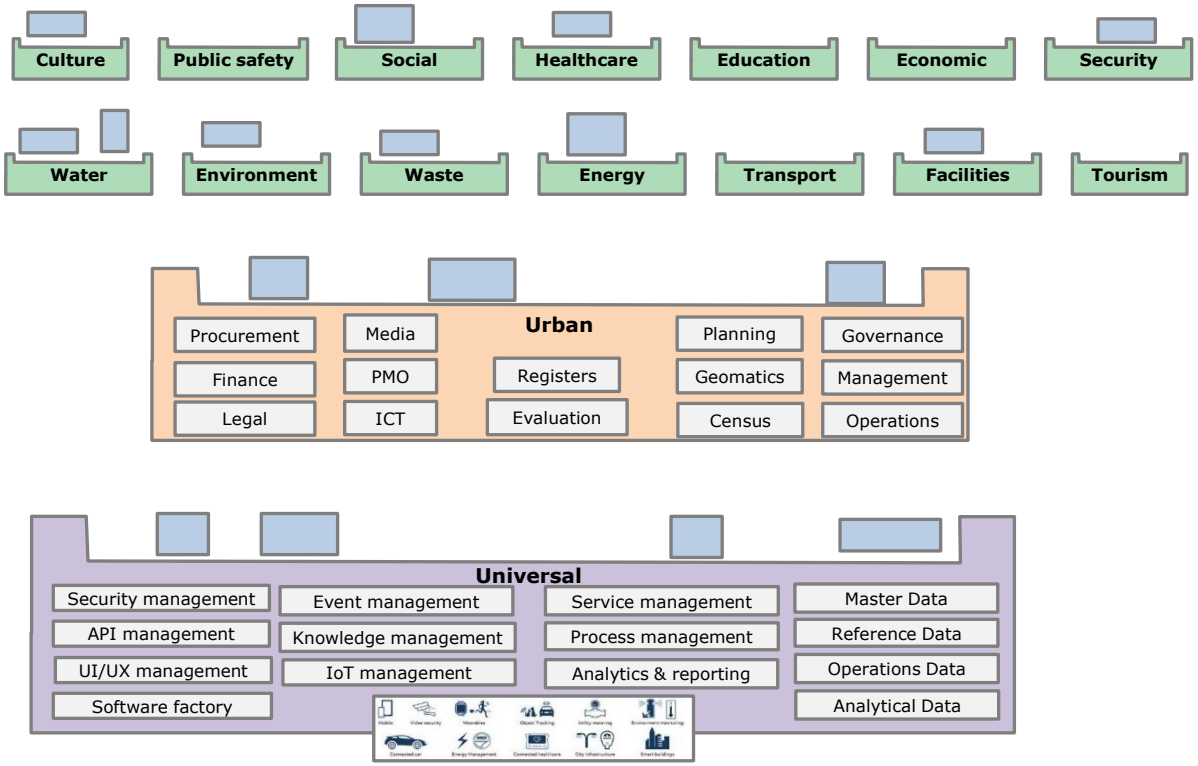
- Stakeholders, their roles and their concerns



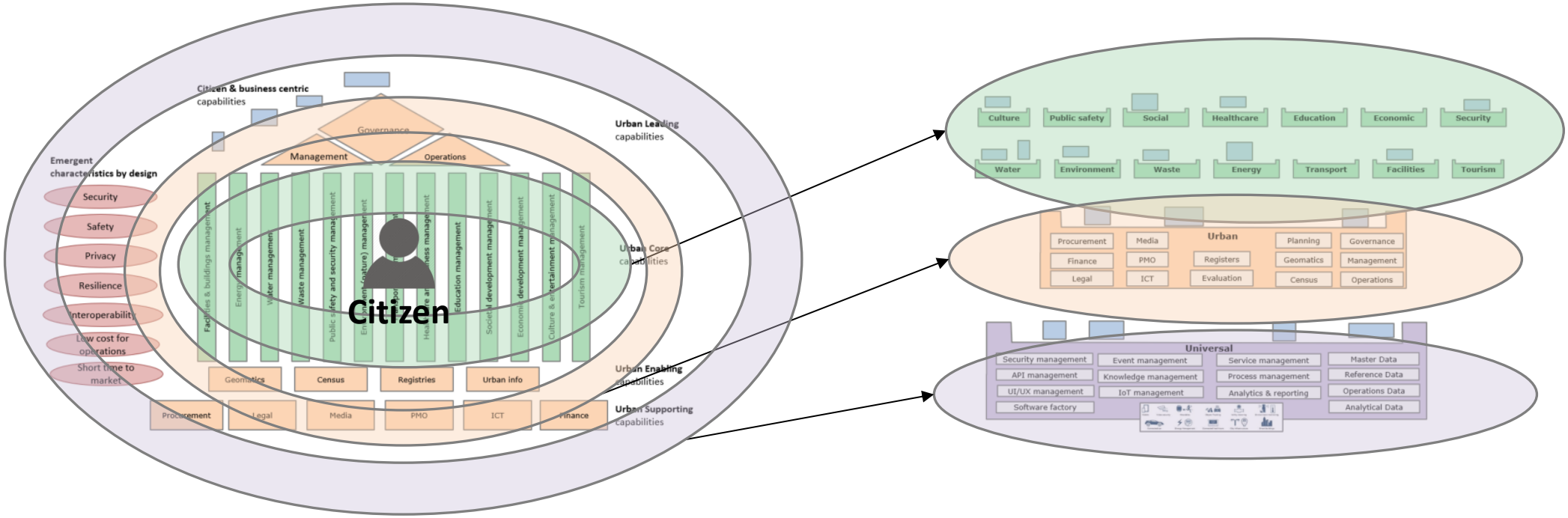
SCRA: Smart Cities reference capabilities



SCRA: Constellation of platforms

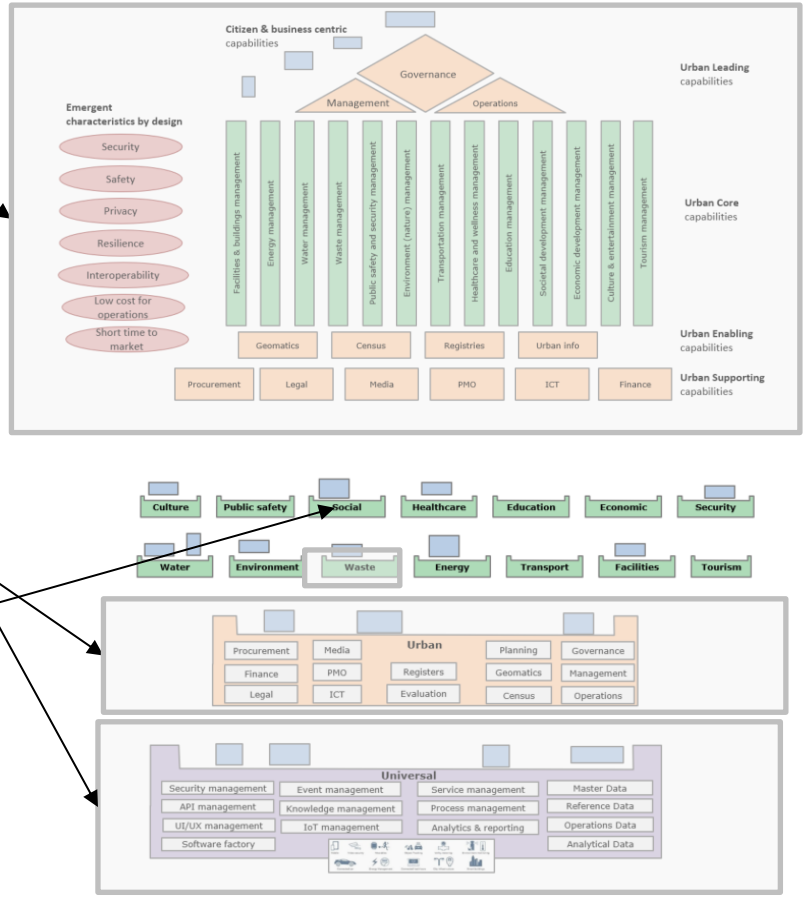


SCRA: Reference capabilities vs platforms



SCRA: Structure

- VALUE view for the whole city
- BIG PICTURE view for the whole city
- SYSTEM-SOLUTION ENGINEERING view for the whole city
- CROSSCUTTING ASPECTS ENGINEERING view for the whole city
- CORPORATE view for the whole city
- RISK MANAGEMENT view for the whole city
- SOFTWARE FACTORY view for the whole city
- STANDARDS view for the whole city
- **ABC** SOLUTION ENGINEERING view
- …
- **UNIVERSAL** PLATFORM ENGINEERING view
- **UNIVERSAL** PLATFORM **ZZZ** COMPONENT ENGINEERING view
- **UNIVERSAL** PLATFORM **YYY** COMPONENT ENGINEERING view
- …
- **URBAN** PLATFORM ENGINEERING view
- **URBAN** PLATFORM **ZZ** COMPONENT ENGINEERING view
- **URBAN** PLATFORM **YY** COMPONENT ENGINEERING view
- …
- VALUE view for WATER vertical
- BIG PICTURE view for WATER vertical
- SYSTEM-SOLUTION ENGINEERING view for WATER vertical
- **WATER** PLATFORM ENGINEERING view
- **WATER** PLATFORM **Z1** SOLUTION ENGINEERING view
- …



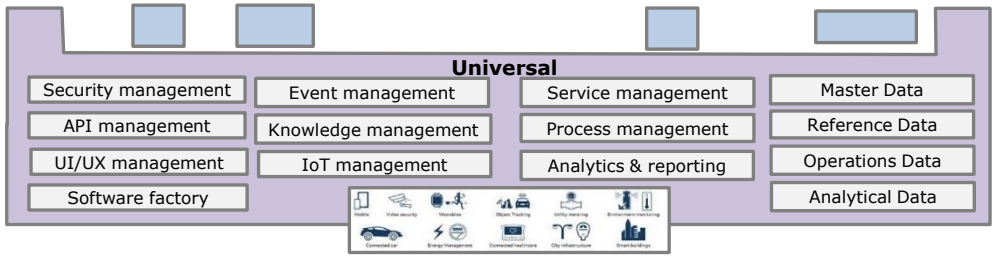
SCRA: Common digital platform (1)

Universal components (tools) of the digital platform

- Reference data management
- Master data management
- Operational data management
- Analytical data management
- Event management
- Information and knowledge management
- Document and content management
- Records management
- Business process management
- Business rules management
- Software factory
- Service and microservice management
- IoT management (following ISO/IEC 30141:2018 - IoT RA)
- Security management
- UX management
- API management

How to standardise?

1. Define necessary capabilities
2. Define APIs to access these capabilities
3. Choose 2-3 products for each tool (low, medium, large)
4. Negotiate one master contract



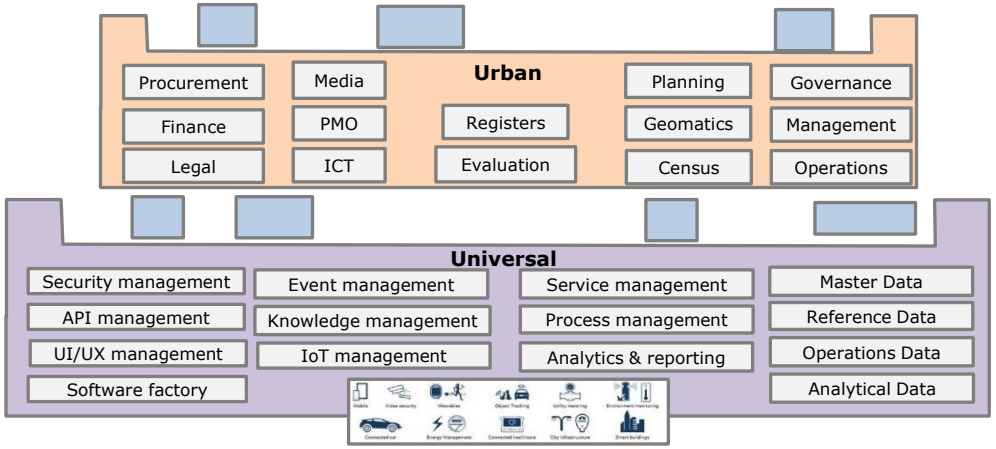
SCRA: Common digital platform (2)

Urban components of the digital platform

- Governance
- Management
- Operations
- Geomatics
- Census
- Registers
- Urban info
- Finance
- Procurement
- Legal
- Media
- PMO
- ICT
- KM

How to standardise?

1. Analyse a city's components
2. Define necessary capabilities
3. Define processes, data, rules, etc.
4. Decompose into services and microservices
5. Establish common design and implementation guidelines
6. Implement as MVP for a first client
7. Improve and enrich with each solution from this domain



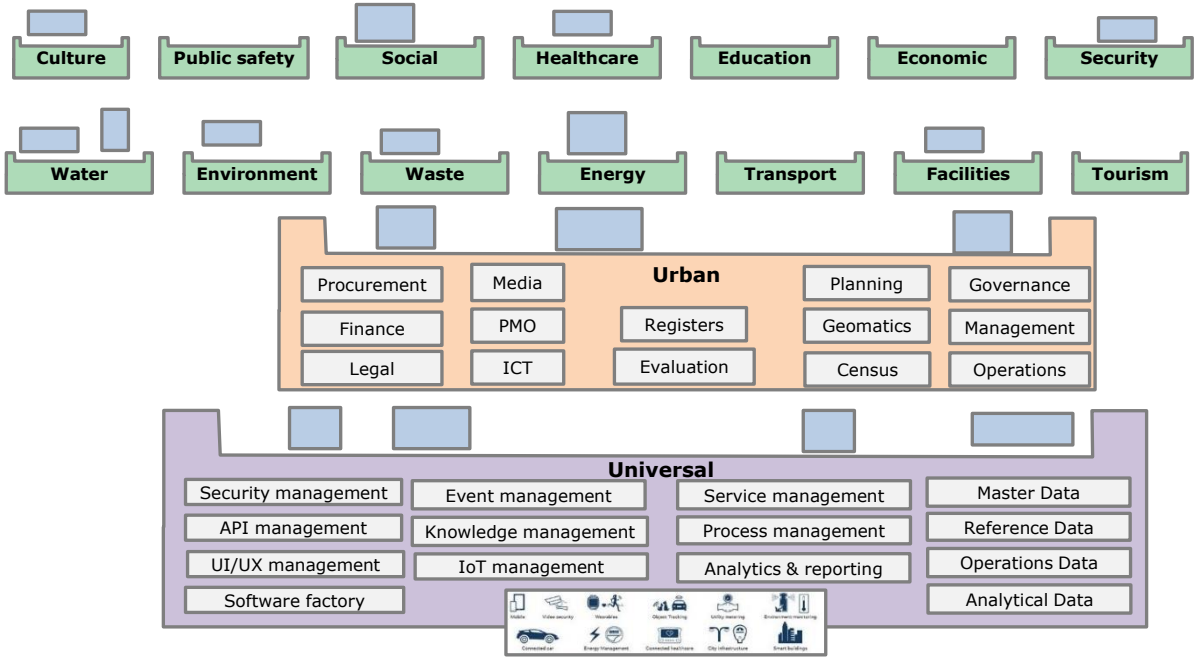
SCRA: Common digital platform (3)

Zone components of the digital platform

- Facilities & buildings management
- Energy management
- Water management
- Waste management
- Public safety and security management
- Environment (nature) management
- Transportation management
- Healthcare management
- Education management
- Social events management
- Economic development management
- Culture & entertainment management

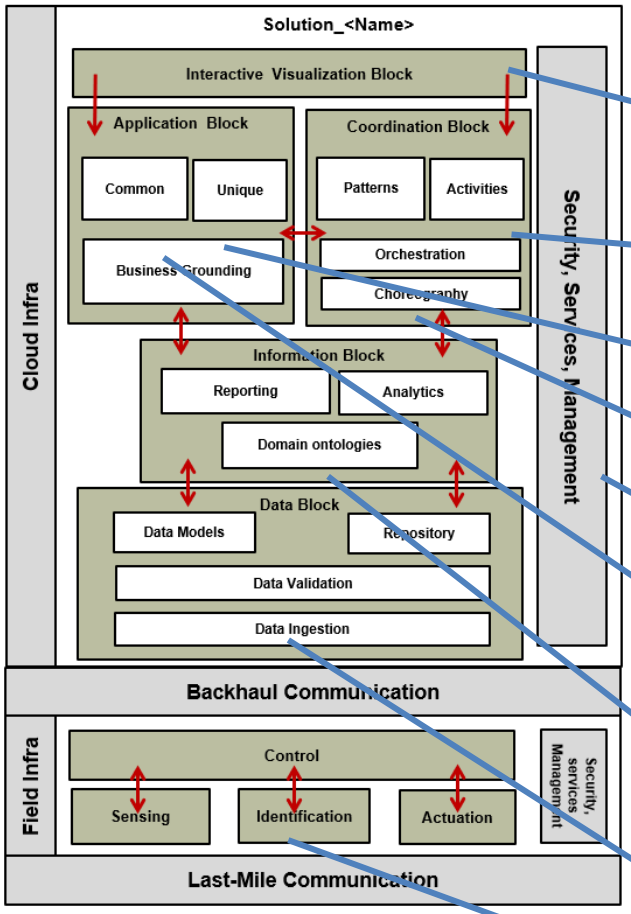
How to standardise?

1. Analyse a domain
2. Define necessary capabilities
3. Define processes, data, rules, etc.
4. Decompose into services and microservices
5. Establish common design and implementation guidelines
6. Implement as MVP for a first client
7. Improve and enrich with each solution from this domain



Platform-enabled agile solutions: reference solution and solution artefacts

Reference solution

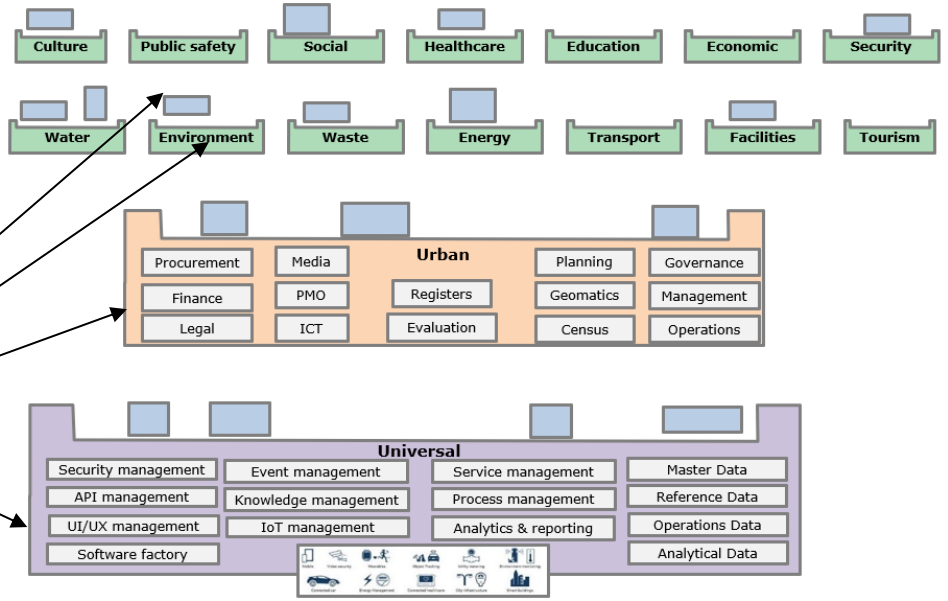


Solution artefacts
Forms
Information, documents
Rules
Processes
Roles
Functions
Information, KPIs, reports, audit-trails
Events, data
IoT

Object

Model

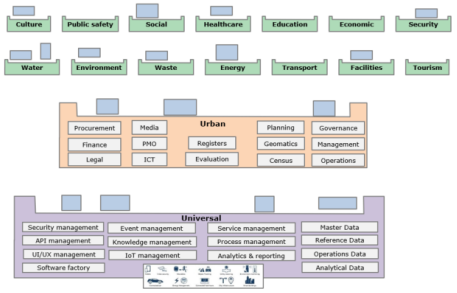
Tool, API

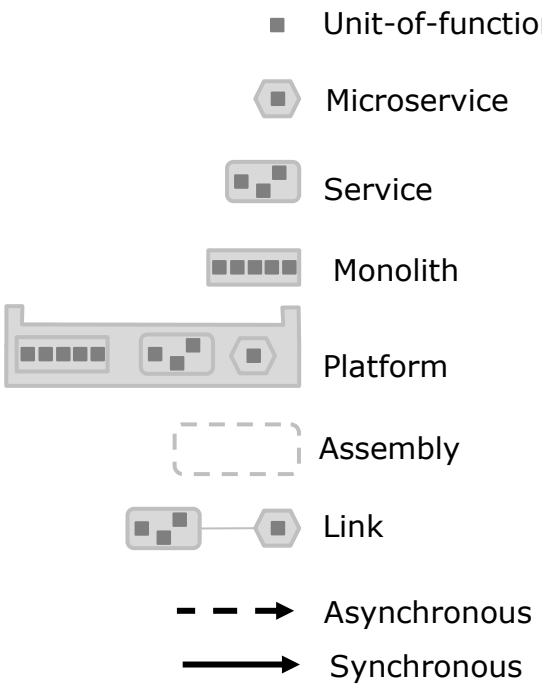


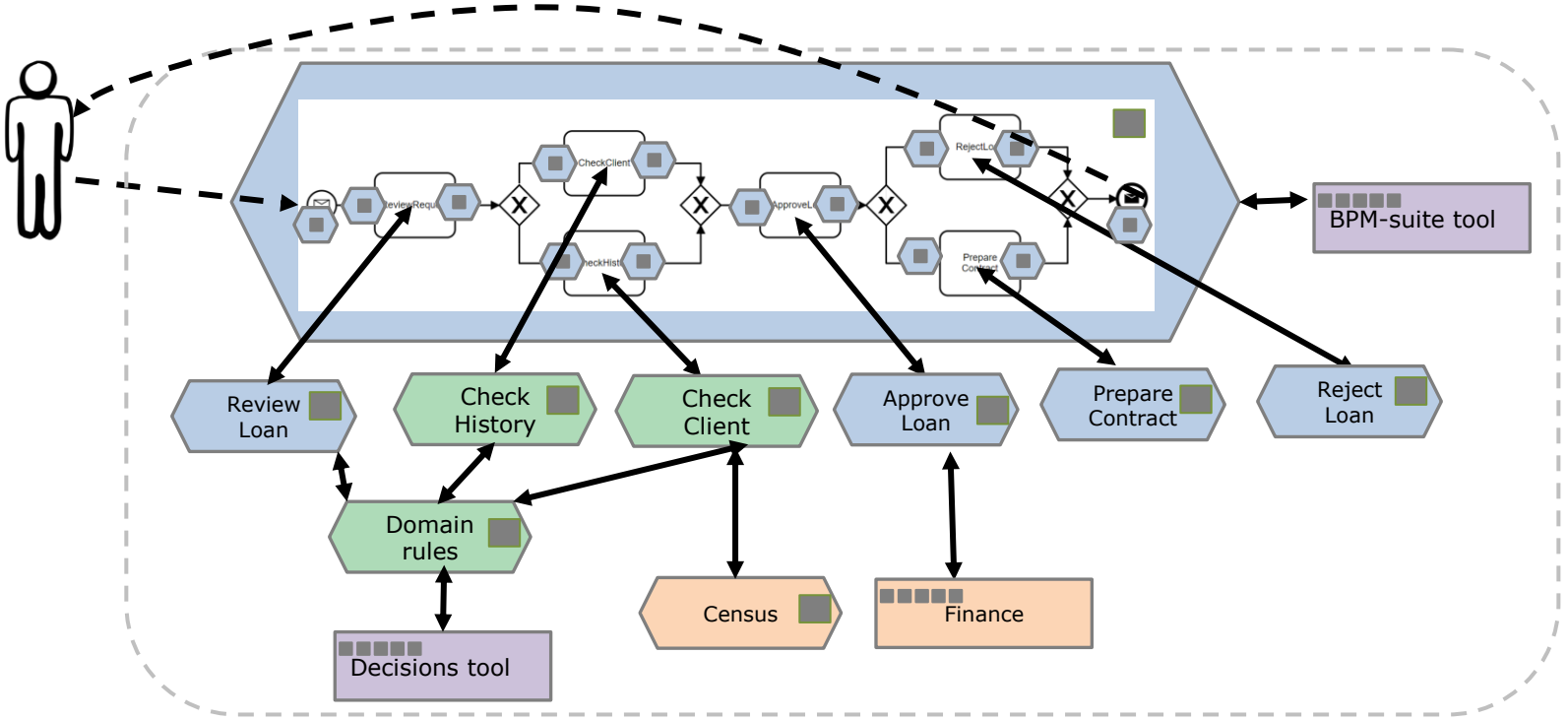
Platform-enabled agile solutions: typology of solution architectures

- An initial set of types
  - event centric
  - data-entry centric
  - document/content centric
  - data and/or information flow centric
  - data and/or information visualisation
  - IoT-device centric
  - mobile centric
  - short-running operations (activities-based)
  - long-running operations (processes-based)
  - any combination
- Each type has its own reference architecture, typical solution artefacts, tools and techniques

Platform-enabled agile solutions: Solution and its microservices



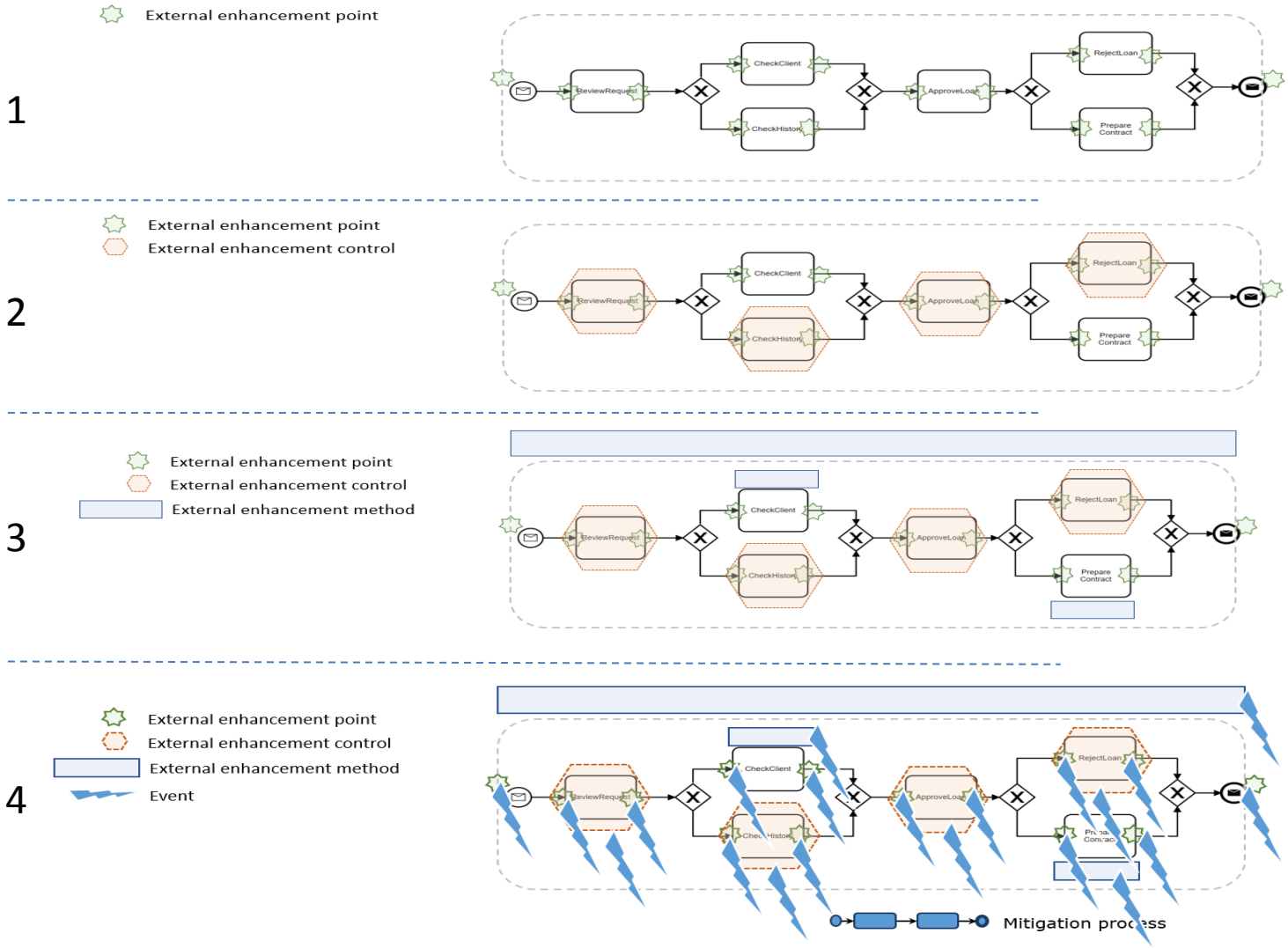




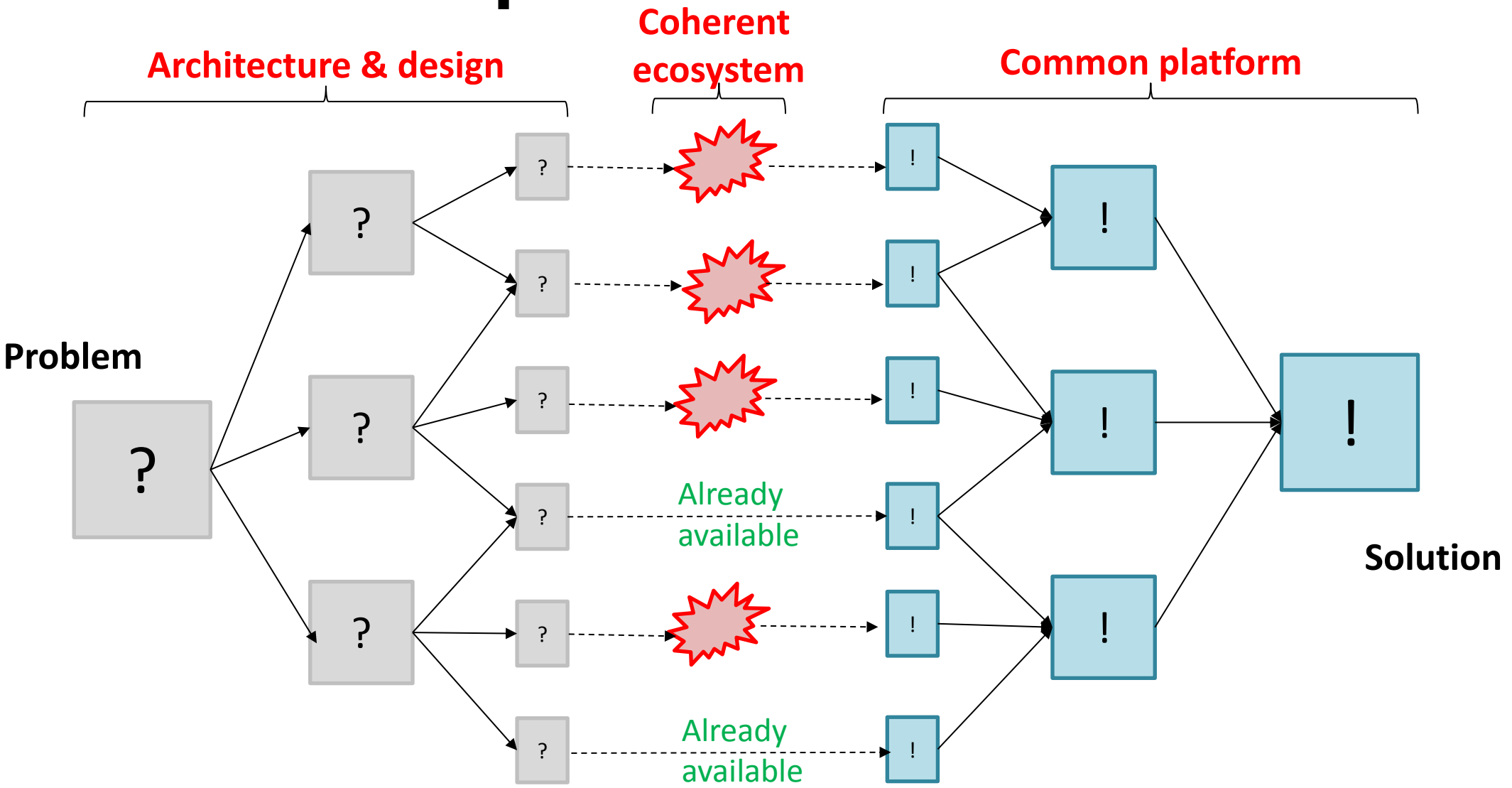
Platform-enabled agile solutions: use of microservices

- Exception: Solution is made only from microservices
- Normal: Solution is made from microservices, services and monolith-supplied functionalities
- Microservices, services and monolith-supplied functionalities are accessible via APIs
- Each API follows common design and implementation guidelines
  - For example, everything is versionable
- http://improving-bpm-systems.blogspot.com/search/label/%23microservice

Security-by-design

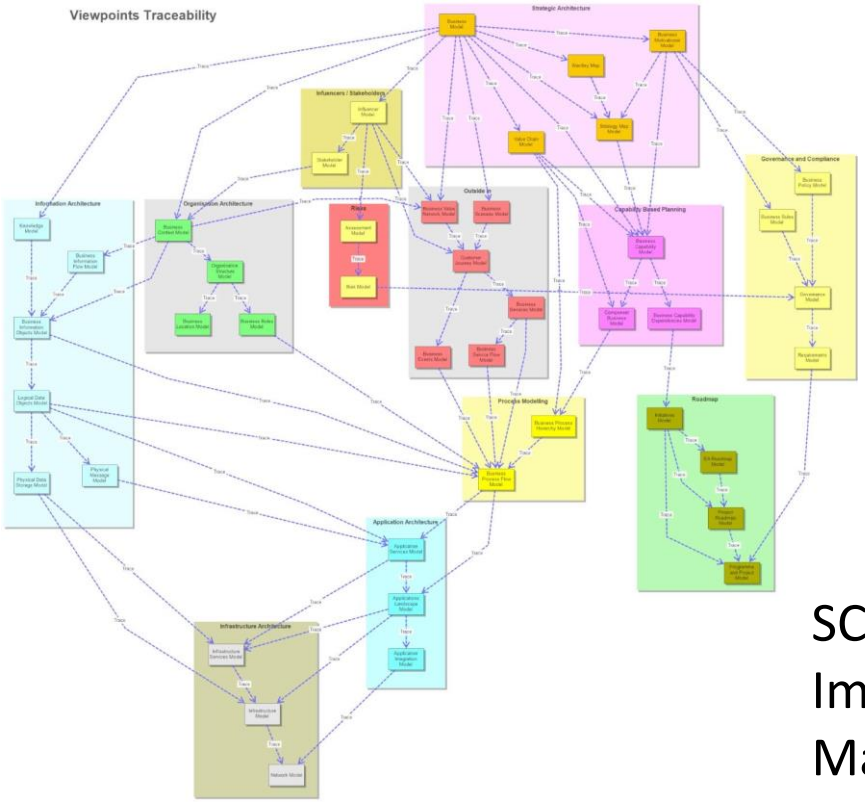


From a problem to the solution



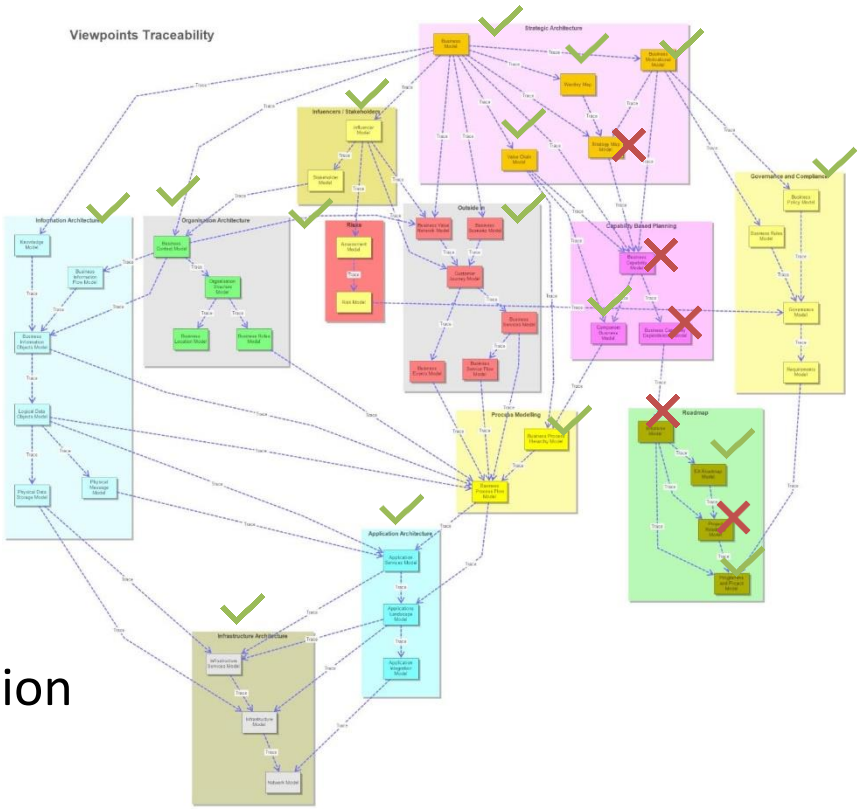
SCRA: Architecture tailoring made easy

SCRA: Reference architecture



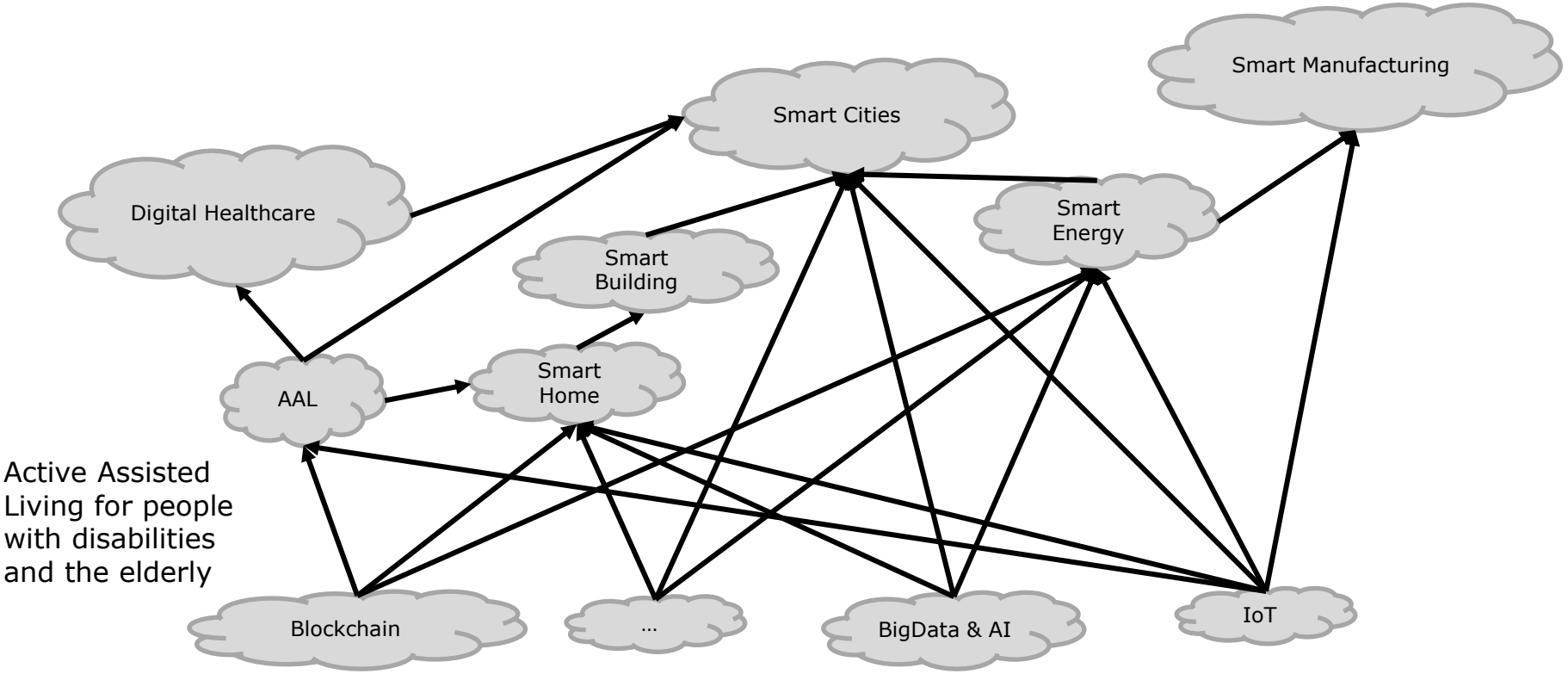
SCIM:
Implementation
Manual

Tailored solution architecture

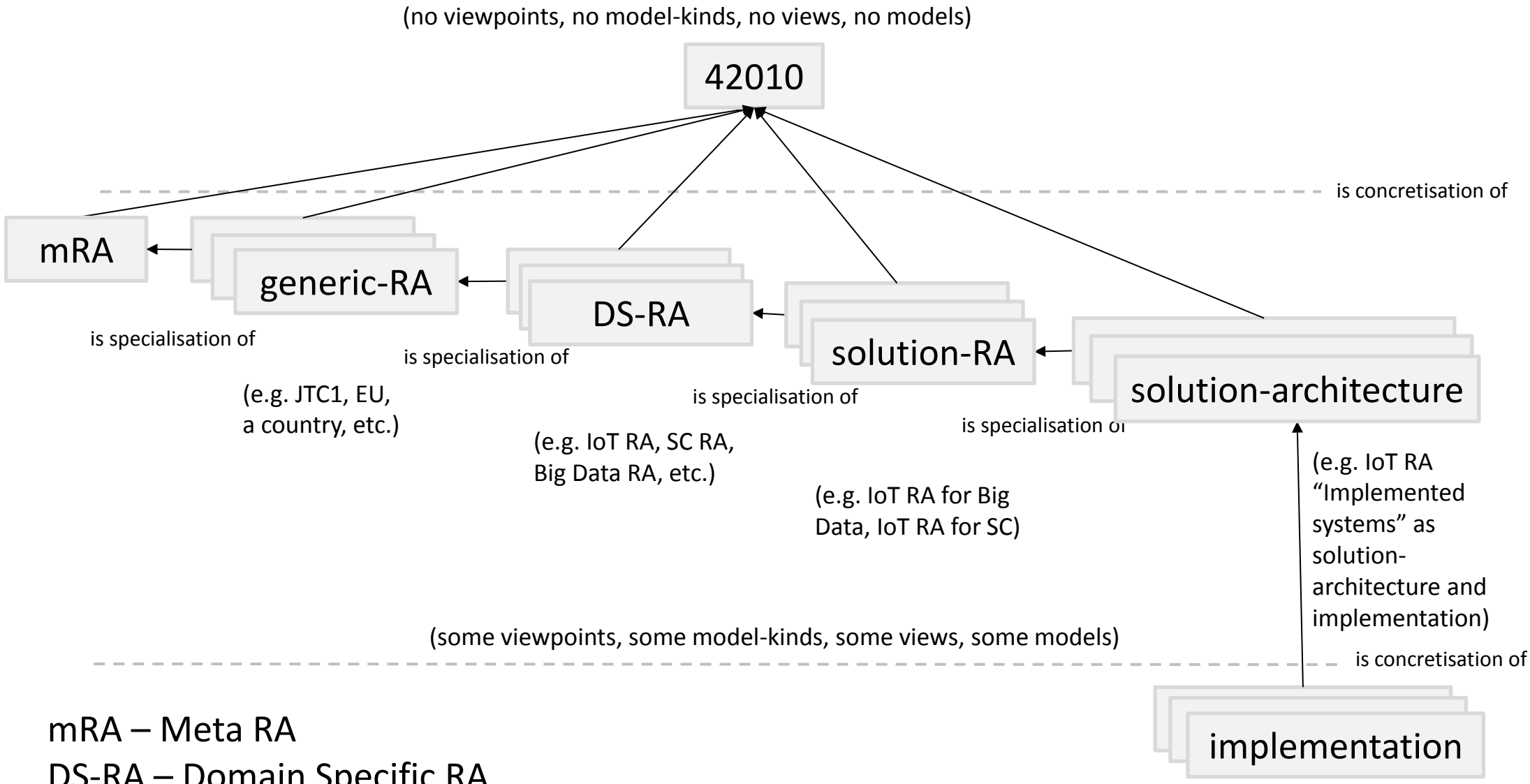


What's about standards?

- Many reference architectures are developed by ISO, IEC, JTC1, IEEE under different methodologies



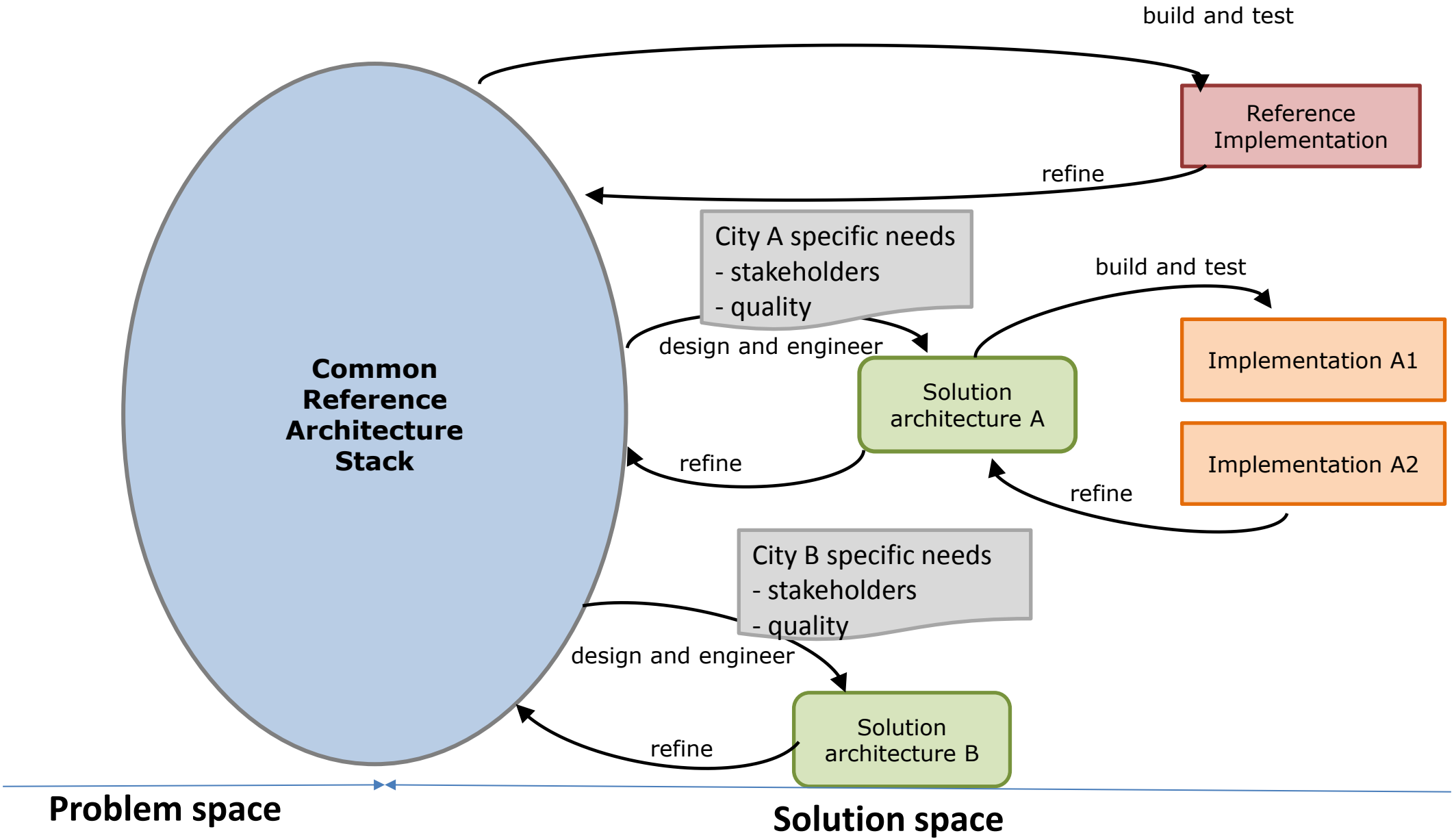
Stack for #ALL system domains



mRA – Meta RA

DS-RA – Domain Specific RA

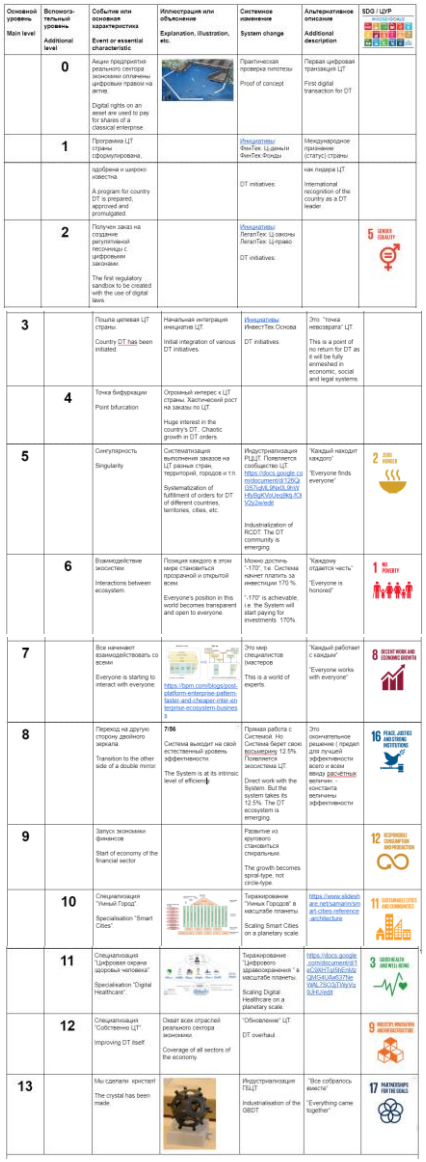
Patterns "Levels of architecting"



Smart City is only a integral part of the bigger picture

- Other parts are
  - FinTech
  - InvestTech
  - MedTech
  - ObraTech
  - UrbaTech
  - ….

Digital Transformation roadmap and maturity matrix



The Digital Transformation maturity matrix is an assessment of the speed of the Digital Transformation roadmap.

Higher level - higher speed. Adding one level increases the speed in 2-3 times. Cost is reducing. Quality is increasing.

Why the speed of DT is very important? Because in the digital world the winner takes all.

Is EA useful for Digital Transformation?

02/09/2019 Gartner Reprint

Priority Matrix for Enterprise Architecture, 2019

benefit	years to mainstream adoption: less than 2 years	2 to 5 years	5 to 10 years	more than 10 years
transformational		**Business Agile** **Digital Business Technology Platform** **EA Governance** **Innovation Management** **Machine Learning**	A Digital Business Continuous Planning ContinuousNext Digital Business Transformation Digital Twin DigitalOps Strategy Realization Office	
high	**Architecture Roadmaps** **Digital Optimization**	Business Capability Modeling Business Ecosystem Modeling Business Ecosystems Human-Centered Design Information Architecture Security Architecture	Business Architecture Customer Journey Analytics Design Thinking Digital Ethics Enterprise Architecture Trendspotting	
moderate	EA Frameworks Solutions Architecture	Agile Architecture Enterprise Architecture Tools		
low				

As of July 2019

Source: Gartner
ID: 369565

Conclusions

- EA is able to
  - solve very complex problems at the scale of Sustainable Development Goals
  - establish a common and efficient set of its tools (frameworks, methods, viewpoints, model-kinds, …)
  - organise concurrent work, coordination, complementarity and reuse (i.e. achieve repeatability)
  - produce digital models
  - make machine-executable enterprises
  - define DT, drive DT and adjust DT as necessary
- **EA is a versatile tool, good investment and strong multiplier of investments**
- **Welcome to the wonderful digital world!**

Questions?

- E-mail: alexandre.samarine@gmail.com
- Mobile: +41 76 573 40 61