



NAP Users Multimodal Forum

NAP Reference Architecture

Harmonization Tool and usage

— with concrete examples of how to use it

Benjamin Witsch-Kovács

NAPCORE X — WG 4.6

15 min · Benjamin Witsch-Kovács

The architecture addresses two NAP types

Metadata Directory

Describes the data and services — the metadata — so consumers can find, assess and subscribe to them.

- Authenticates data providers, lets them insert metadata records
- Search & discovery for end users; machine-readable feed to other portals
- Not involved in the actual data exchange

Data Platform

Holds the metadata and the data / services themselves — the NAP becomes an intermediary.

- Authenticates providers (optionally consumers); ingests data & metadata
- Adds distribution interfaces for data dissemination
- Data still remains owned by the data provider

Static data store — a light middle option: a Metadata Directory that may also store some static, open-access data.

Benefits of a harmonized NRA

1 Common understanding

Defined NAP types and shared terms mean everyone refers to the same things when they talk about NAPs.

2 Clear expectations

Clear dissection of a complex system into simpler views each describing one facet. Eg. functional diagram with dataflows described in detail..

3 Better discovery

Agreed dataset descriptions — metadata — with naming and keyword guidance greatly improve search, especially on combined NAPs.

4 Technical understanding

Specified technical interfaces let a service provider reach data from multiple NAPs through a single framework.

Use cases of the architecture

1

Build a harmonized NAP from scratch

every technical and organisational element needed for an operational NAP — a complete blueprint

2

Harmonize an existing NAP

adopt existing NAPs to the reference architecture, step by step, on a basic level

3

Benchmark and compare NAPs

a basis to compare NAPs and their complex parts, aligned with the KPIs and the NLKF

The architecture also carries its own sources

Five views describe the NAP. Three supplement packages say where the description comes from — and make every statement traceable back to a document.

Specifications & Recommendations

The formal sources the architecture must respect — legal acts, technical standards, data-model profiles and NAPCORE's own recommendations.

- 16 legal acts and 19 technical standards
- 5 data-model profiles
- 5 NAPCORE recommendations

40 specifications

Requirements

The bridge. A requirement turns a clause into a concrete, model-relevant obligation and attaches it to an architectural object.

- derived from a specification or a KPI
- attached to a subsystem, module, interface, actor or function
- the only link between the two worlds

the one bridge

NAP Level of Service KPI Framework

The KPI framework modelled inside the architecture, so a change on either side is visible on the other.

- 7 feature categories as folders
- each KPI is a specification object
- 2024 release embedded in the model

43 KPIs

What FRAME gives us — and what NAPCORE added

The five NRA views

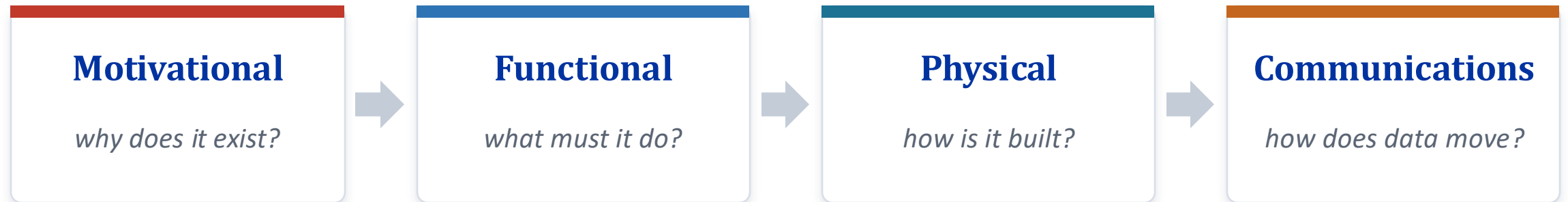
- 01** User Needs / Motivational
- 02** Functional
- 03** Organisational
- 04** Physical
- 05** Communications

NAPCORE added

- Mission, vision and the ITS service definition for a NAP
- Two NAP types elaborated in parallel through every single view
- The physical, communications and organizational views, filled in for the NAP
- Specifications, requirements and the KPI framework as modelled packages
- *Safety and security as a cross-cutting concern — still work in progress*

The skeleton is FRAME's, the NAP content is ours. Where FRAME did not stretch far enough, the extensions are documented as methodological additions rather than hidden.

Each view answers the question the last one raised



User needs sit between the first two — a function exists in the model only because a user need makes it necessary.

Organizational — who is responsible? Cuts across all four: every function, module and interface has an actor behind it and, usually, an agreement.

Safety & Security — what can go wrong? Also cross-cutting, and honestly the least finished part of the model — we come back to it near the end.

Because the views are linked, a change in one is traceable into the others — that is the whole point of modelling rather than writing prose.

What interoperability really takes

Interoperability is not one document — it is three habits, aimed at four audiences.

1 Talk the same language

shared terminology and typology

2 Use the same standards

common rules for data exchange, structure and content

3 Follow best practices

recommended look, feel and behaviour

For whom, and why

Decision makers

identify key concepts — via terminology

Operators

run the system as recommended — metadata, naming

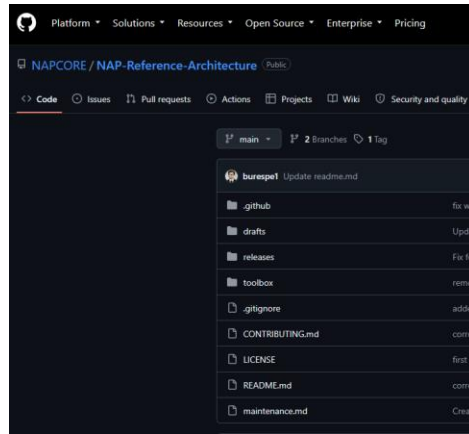
Developers

implement interoperable features — via interfaces

Subscribers

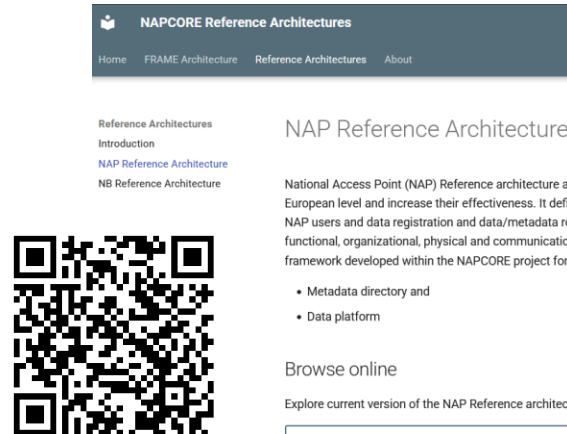
follow the example of a first subscription

Where to find more information



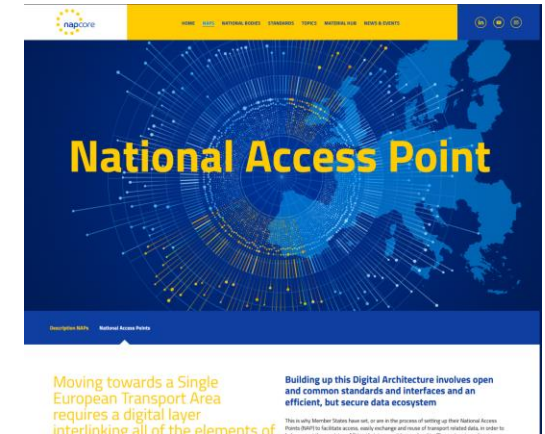
GitHub repository

the model, the documentation and
the release history
github.com/NAPCORE/NAP-Reference-Architecture



Reference Architectures site

the views, browsable, without
opening a modelling tool
napcore.github.io/Reference-Architectures-website



NAPCORE

the project, the working groups
and the deliverables
napcore.eu

Also: the NLKF annual assessment report (per-NAP results and gaps) · the NAPCORE recommendations · a journal paper describing the NRA and the NLKF together, currently under review.

WHERE WE ARE

Conclusions & Contact



- **NRA published in v1.0**
the reference architecture is out and in use
- **Review on every view ongoing – preparing v2.0**
each view is being refined — your input matters
- **Aligning with other NAPCORE tasks**
ecosystem architecture, including security

Questions & Contact

Benjamin.witsch-kovacs@austriatech.at
petr.bures@cvut.cz