

Full workflow to a better form

openEHR basics

openEHR – what's that?

- Healthcare IT standard that provides specifications of **how to store, share and retrieve health data** -> principal idea of **separating this data from applications**

MOTTO

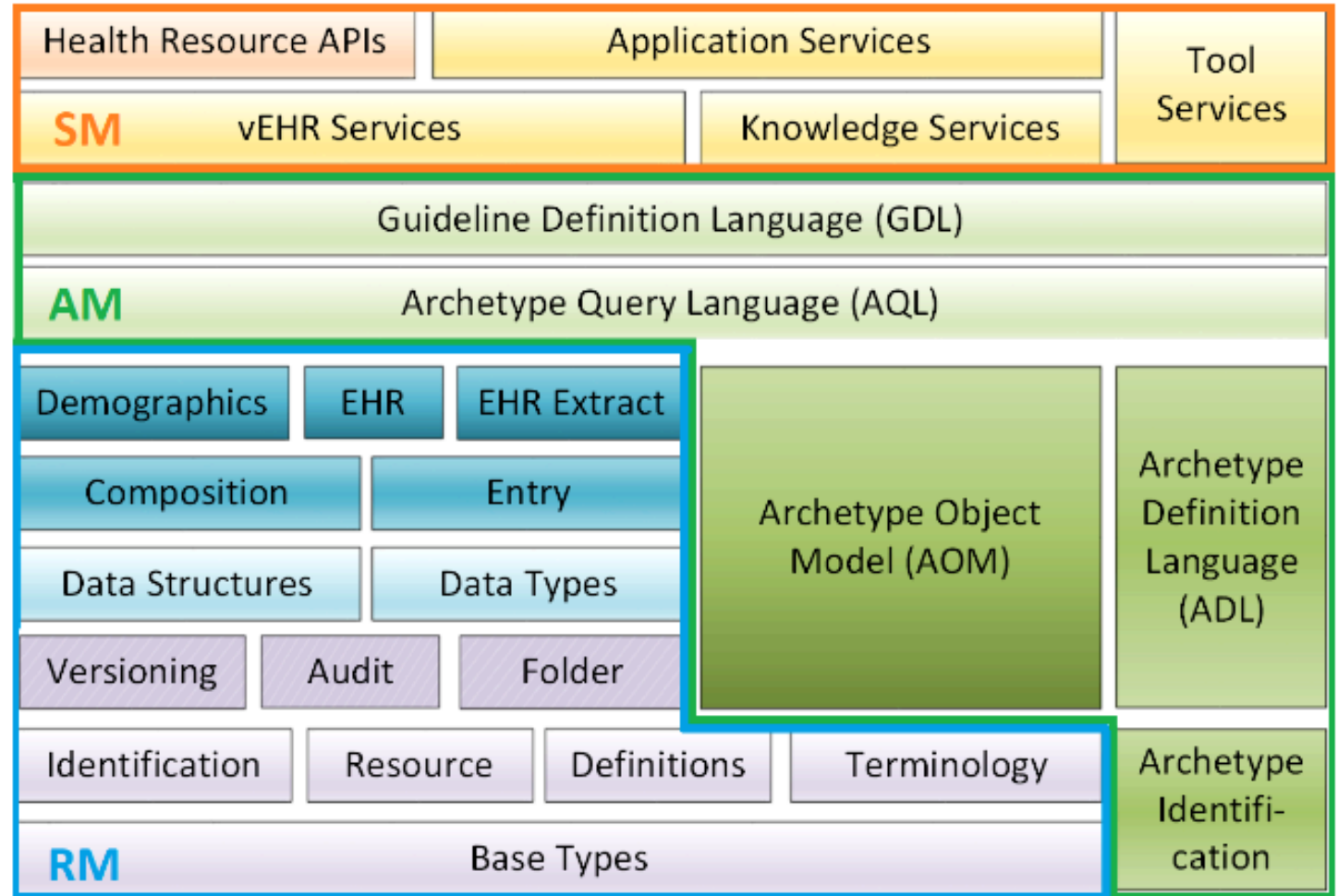
The transformation of health data from physical format to electronic format -> thus **ensuring universal interoperability** between all forms of electronic data

openEHR - benefits

- Working with medical information at **the same semantic level**
- Lowering the problems with **mismatched clinical information models**
- **Strengthening the interoperability** and making possible the use of **analytic functions** -> such as research querying and decision support.
- **Separation** of patient **personal information** & demographics from **clinical data**

openEHR – a bit technical part

- **Reference Model (RM)** and **Service Model (SM)**: root and basis of the system - **THINKEHR**
- **Archetype Model (AM)**: has the content for clinical or administrative data - **ADL DESIGNER + THINKEHR**



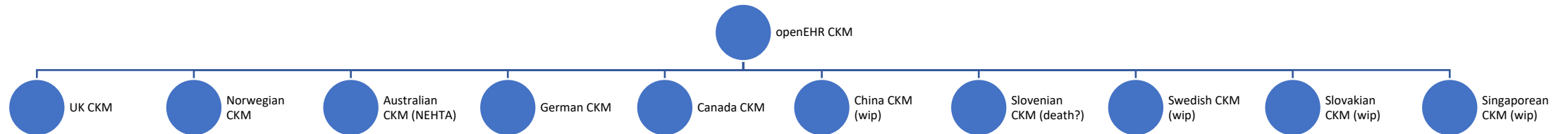
openEHR - artifacts

- **Archetypes** – Structured clinical concept that can be understood by a machine.
- **Templates** – Set of archetypes that can be understood as a clinical form.
- **openEHR CKM** – International repository of archetypes and templates.

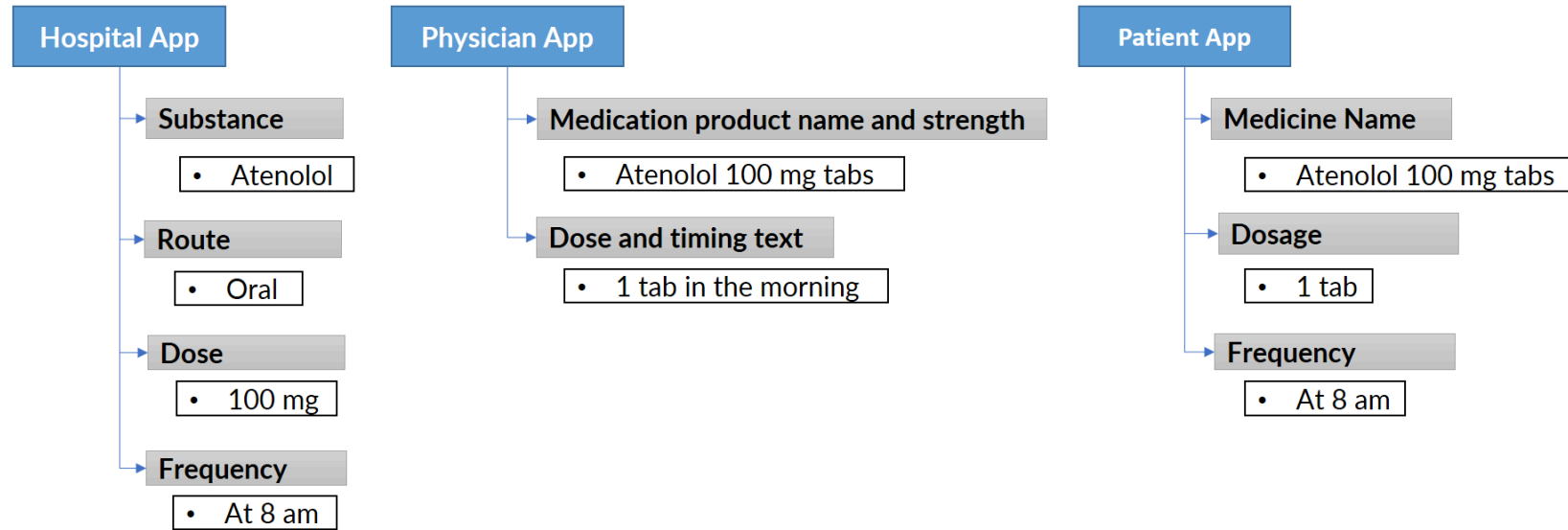


Lego analogy for openEHR resources

International CKM's available



Clinical models issue



Using openEHR resources, this dilemma is easily solved if using the archetypes from CKM, because all the information will be saved or retrieved in exactly the same way on each application.

Maximum data set dilemma

- “Sometimes we can not find our use case on openehr ckm!”
 - Did you **share your use case or suggest** with the openehr ckm team your issue?
 - Did you search it on the other available ckms?
 - Are you really trying to save clinical data or confusing the concept?
 - Did you start your search with enough time for it? **Modelling usually takes more time than the normal development of applications.**
- If none of these cases were checked, openehr ckm team **will not care about your use case** – they can not discover and are not obligated to know in what you are working on, unless you start to share your issues with them.

Modelling – why should I use archetypes from openEHR CKM?

- Using **same** archetype from CKM ensures interoperability between systems – if in your server there's an archetype (embedded in a template) that speaks language X (blood_pressure.v2), he will know how to speak with other server that has the same archetype (blood_pressure.v2) and the data can be queried using the exact same AQL.
- When we speak about an archetype from openEHR ckm, it must have never been changed (anything on his content) -> if an archetype from openEHR ckm is changed, integrity is lost along with the interoperability possibilities – his language has been changed.

Modelling

- Creating an archetype takes **WAY MORE** time than creating a template. If we need to create a **proper** internal archetype, it grows exponentially.
- openEHR modelling *joke*: “If you are doing it quickly, for sure, you are doing it wrong”.
- Archetype - similar as business logic along with data formats specification:
 - Eg: Blood pressure – has systolic pressure with:
 - Type: quantity
 - Unit: mmHG
 - Decimal cases / precision : 3
 - Range/magnitude: <1000
 - Internal connection to terminology: SNOMED-CT
 - Language: english

Terms !

ADL –
Archetype
definition
language

AQL –
Archetype
query language

OPT – template
paths

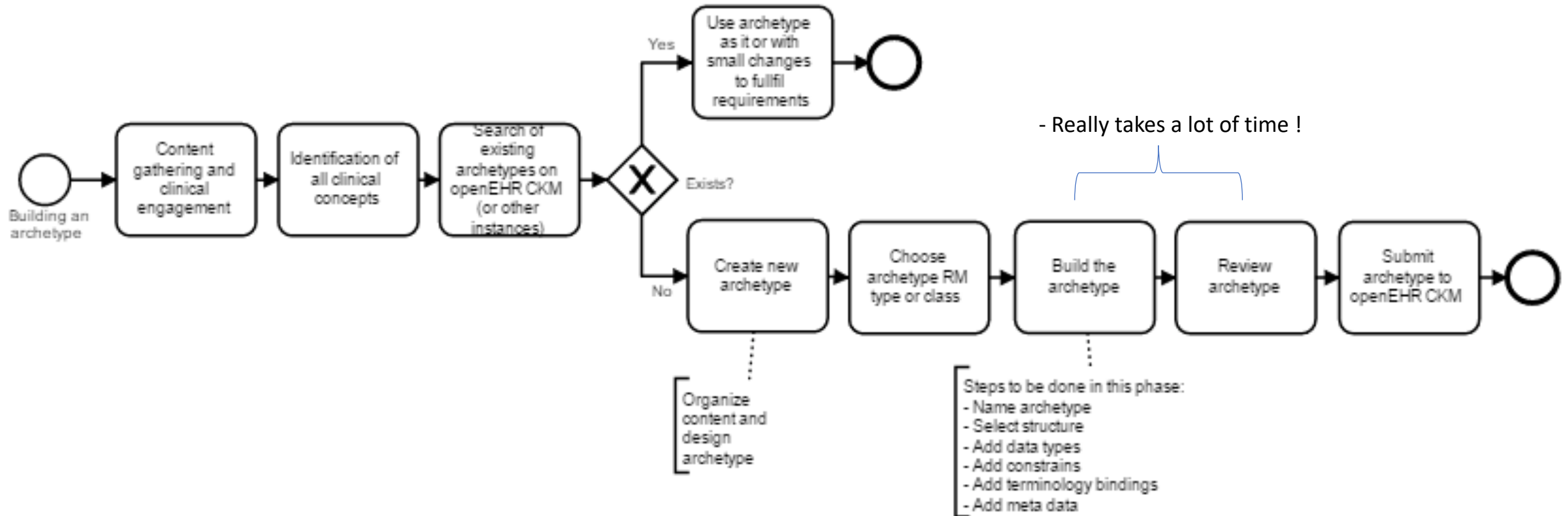
Blood Pressure (Latest revision / latest published)

```
definition
  OBSERVATION[at0000] matches {    -- Blood Pressure
    data matches {
      HISTORY[at0001] matches {    -- history
        events cardinality matches {1..*; unordered} matches {
          EVENT[at0006] occurrences matches {0..*} matches {    -- Any event
            data matches {
              ITEM_TREE[at0003] matches {    -- blood pressure
                items cardinality matches {0..*; unordered} matches {
                  ELEMENT[at0004] occurrences matches {0..1} matches {    -- Systolic
                    value matches {
                      C_DV_QUANTITY <
                        property = <[openehr::125]>
                        list = <
                          ["1"] = <
                            units = <"mm[Hg]">
                            magnitude = <|0.0..<1000.0|>
                            precision = <|0|>
```

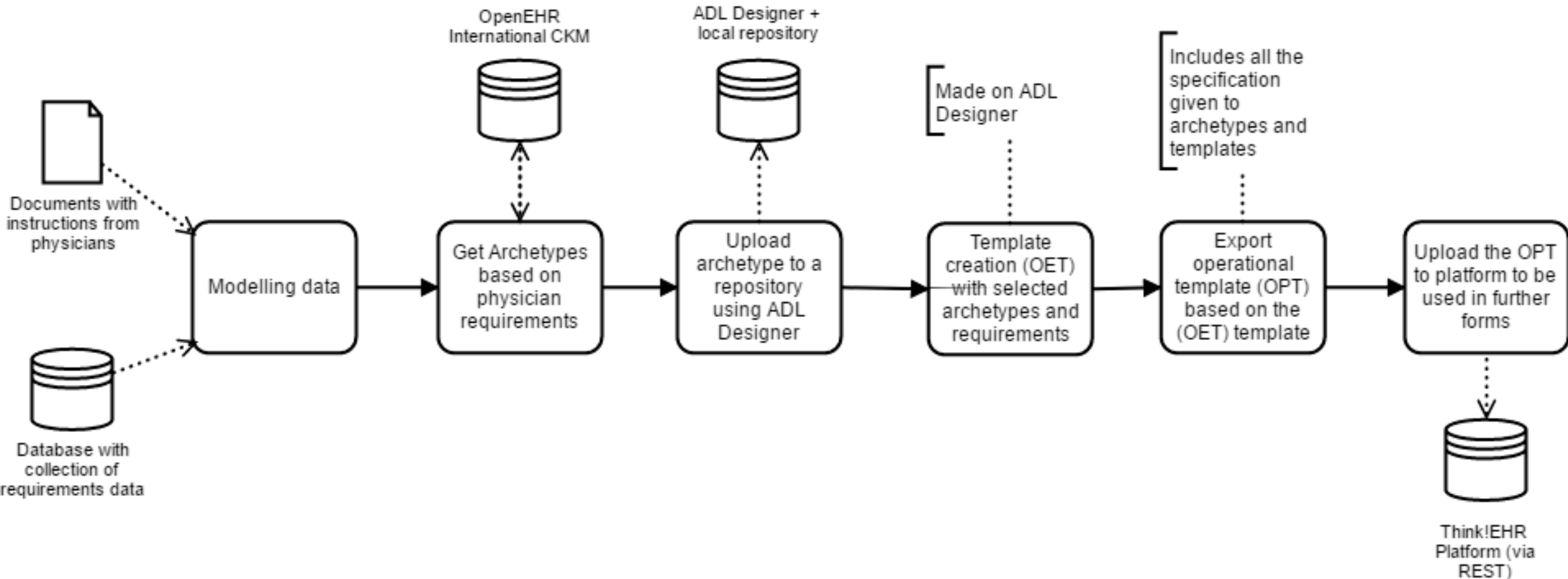
```
select
e/ehr_id as EHR_ID,
a_a/data [ at0001 ] / events [ at0006 ] / data [ at0003 ] / items [ at0004 ] / value as Systolic ,
a_a/data [ at0001 ] / events [ at0006 ] / data [ at0003 ] / items [ at0005 ] / value as Diastolic
from EHR e
contains COMPOSITION a
contains OBSERVATION a_a [ openEHR-EHR-OBSERVATION. blood_pressure . v1 ]
where
a_a/data [ at0001 ] / events [ at0006 ] / data [ at0003 ] / items [ at0004 ] / value /magnitude>=180 or
a_a/data [ at0001 ] / events [ at0006 ] / data [ at0003 ] / items [ at0005 ] / value /magnitude>=160
```

```
"vital_signs_pathfinder_demo/blood_pressure:0/any_event:0/systolic|magnitude": 2,
"vital_signs_pathfinder_demo/blood_pressure:0/any_event:0/systolic|unit": "mm[Hg]",
```

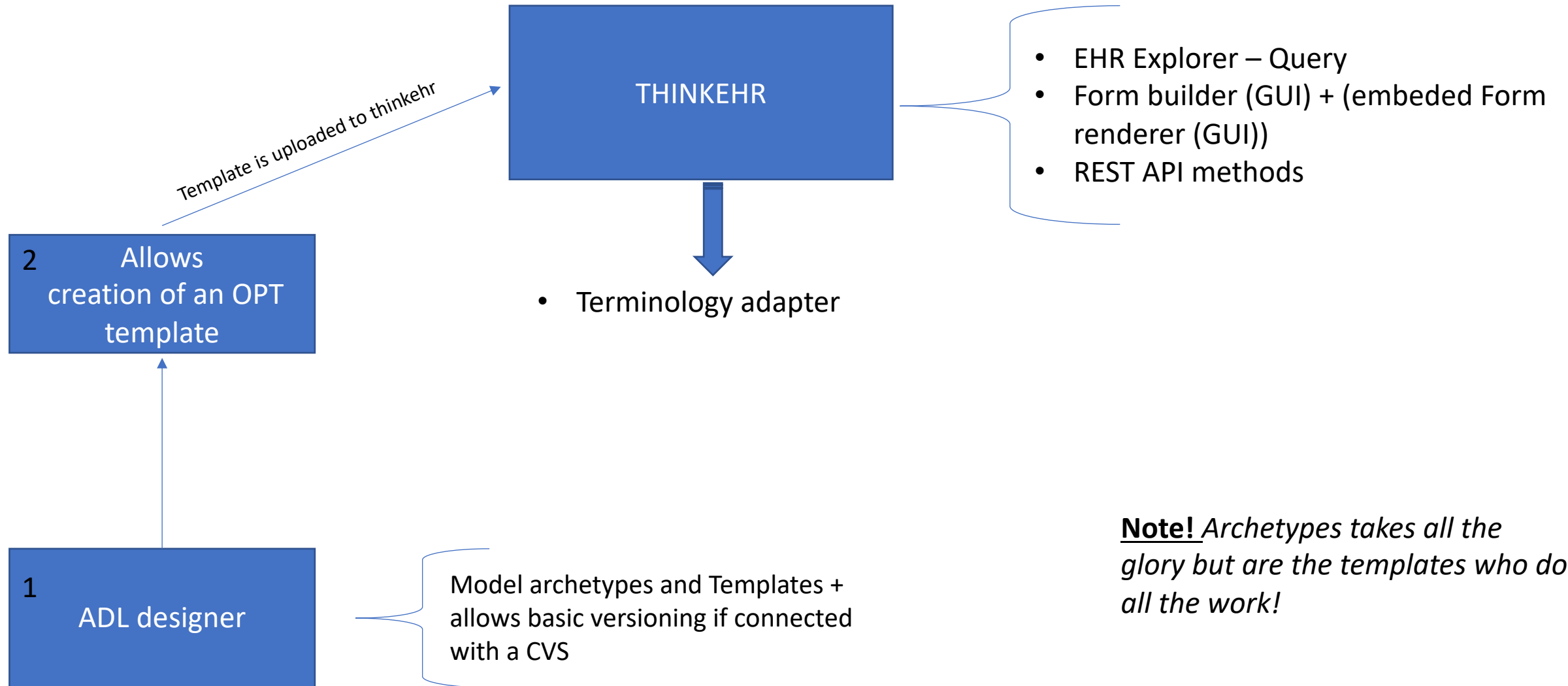
Modelling - Archetype Development Process



Workflow from modelling until start using THINKEHR



ThinkEHR to Marand tools



Note! Archetypes takes all the glory but are the templates who do all the work!

ThinkEHR Forms - FormBuilder

- Form builder is a GUI on top of the openEHR Templates.
- It works using template paths:

```
"vital_signs_pathfinder_demo/blood_pressure:0/any_event:0/systolic|magnitude": 2,  
"vital_signs_pathfinder_demo/blood_pressure:0/any_event:0/systolic|unit": "mm[Hg]",
```

The screenshot displays the ThinkEHR FormBuilder interface. On the left, the 'Registry Information' panel shows a 'Document type' dropdown set to 'SNOMED-CT(2019)::721914005::...' and a 'careflow_step' dropdown. On the right, the 'Details' tab is active, showing the 'Template path' as 'patient_consent/context/xds_metadata/document_type'.

- If you change the name of these paths on the template, the form stops to work – we will see this on the demo.

openEHR specification versions between marand tools

- Latest openEHR specification: Z
 - ThinkEHR is based on version X (aprox 2 years old and with tweaks – plan final release until final of this year)
 - ADL designer is based on version Y
 - Both tries to work together to map each other, no issues found in the last 2 months.
- Outsider modellers are not aware of this and work on the latest openehr specification.

Form builder & Form renderer

WEB

FB:

- No retro-compatibility from the previous form builder present on EHR explorer, all forms had to be rebuild – Pathfinder case results on additional 1 month of work just to adaptation and understanding the new FB along reporting of missing features.
- Major version is changed every 2 months – still in development.

FR: gives the style to the FB and Pathfinder FE

- Minor versions changing every week, some forms stop to work if require dependencies from API, some terminologies could stop to work too.
- Major version is changed every month – still in development.

- All these changes are necessary + it is good to be aware this process.

IOS:

No idea of what's the current implementation – it's not compliant with changes on web.

Form builder

- Benefits – allows quick building of forms on top of openEHR templates (each drag n drop is a piece of code added to the final form) along with immediate deploy – this means that a form that took 1 month to be done, can be done in approx 1 week (depending on the complexity) and ready to use on all other thinkehr instances with combo (template+form).
- For a proper form, additional work needs to be done – it's necessary to understand how the template was done to use for example dependencies or terminologies.
- Can only be used for openEHR formats.
- Not yet prepared for horizontal forms – e.g. questionnaires.

Form renderer

- The versions update can make terminologies, dependencies or visuals to change – check NHS forms on openpath vs pathfinder-lite.
- Sometimes who made the update is not aware of these cases.

Extra - Who commands on this workflow?

1º Archetype

2º Template (if we allow this to happen on the archetype – e.g. change constraints for each attribute)

3º Form (if we allow this to happen on the template)

As a end user never forget: **Every change on a template is reflected in the form! They work together.** Nevertheless a change on a form is doable if the template allows it.

Main mistakes found on analysis of customer&marand produced templates and archetypes (1)

- 1. Constrain of overall template occurrences and misunderstanding of the concept
- 2. Wrong usage of the ADHOC archetype (usually same issue above)
- 3. Manually edited archetypes from openEHR CKM
 - 3.1 – internal atcodes changed
 - 3.2 – internal terminology changed
- 4. Outdated versions of openEHR CKM archetypes being used (logic code can be wrong and had be fixed on CKM)
- 5. Lost track of archetypes that were used to create a template – no management and versioning of openEHR artifacts -> catastrophic in a production environment.

Main mistakes found on analysis of customer&marand produced templates and archetypes (2)

- 6. Wrong usage of COMPOSITIONS types to define a template
- 7. Wrong usage of RM types (ACTION, OBSERVATION, INSTRUCTION, CLUSTER, SECTION, etc.) on the internal archetypes
- 8. Existence of archetypes with information that's not meant to be saved as clinical data – misunderstood of the concept?
- Conclusion: Were the responsible persons for taking care of these cases really prepared and had the knowlegde to deal with this?

You don't put a kitchen chef taking care of an restaurant energy suply
or programming the restaurant DB, right?
Why do it with openEHR then?

Latest issues

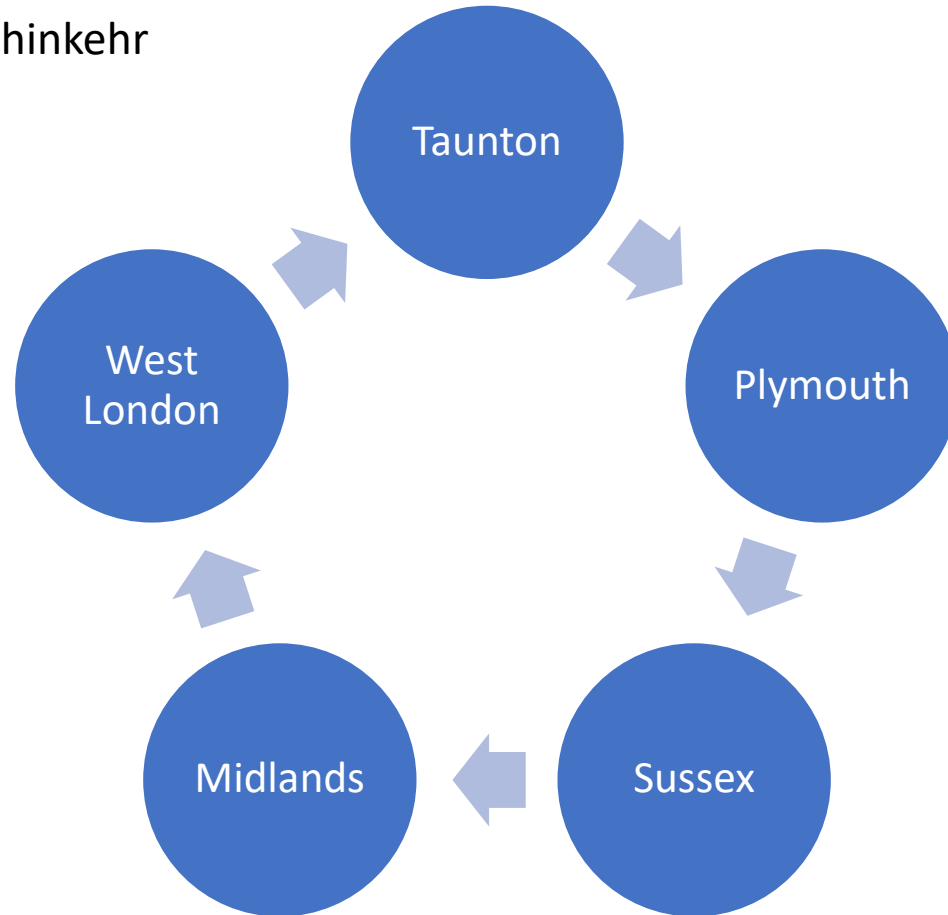
How should we deal on a production environment with:

- Versioning an archetype?
- Versioning a template?
- Versioning a form?

Changing Templates on production

Using NHS use case:

- Each trust with 1 thinkehr or a shared thinkehr
- MUST template + form
- Trusts share data between them



FAQs

I changed an archetype and I cannot see any change on my template.

- Any change on an archetype needs a partially rebuild from all the templates that make use of this archetype. If you change an archetype that is used in many cases, think in advance what can occur and do it wisely.

When opening a template on ADL Designer, I got an alert about a hash that was changed, why?

- Every archetype and template are digitally signed with an MD5 hash, that reflects their content integrity. If any artifact was changed, you will be notified with this message – be careful, since any save after this can provoke your template to malfunction. In cases of openEHR repositories being maintained by many people it can happens that someone changes an archetype and it is not aware of all the templates that use this archetype. If you are not aware of this change, do not save the template again.

FAQs

I changed an archetype, update the template and my form does not work.

- Any change on an template needs a partially rebuild from all the forms that make use of this template. If you change an template that is used in many cases, think in advance what can occur and do it wisely.

Demo

- OpenEHR CKM
- ADL designer
- ThinkEHR
- Terminology adapter - terminologies
- Form builder + form renderer
- Pathfinder-lite (without tasks motor - camunda)
- Archetype, template and form: MUST
- Complex case: Comorbidities or Problem list