Paper SI06

Creating CORE Rules from Biomedical Concepts

Johannes Ulander, data4knowledge, Umeå, Sweden

ABSTRACT

The work on improving our clinical data standards to utilize modern web development stack and graph technology is happening. It impacts how we access and manage standards, providing better control and opportunities to change the data flow for collecting and reporting clinical data. Now that the CDISC Open Rules Engine (CORE) and the first Biomedical Concepts are available from CDISC, it is time to start using them to take the next step in improving the quality of the data delivered in clinical studies and easing the burden of QC for programmers.

This presentation will go through the steps used to automatically create Core rules matching Biomedical Concepts accessed via the CDISC Library API and show how easy it is to start doing this on your own.

It will show how this also impacts how we automate creation of forms and discuss learnings and future improvements.

BACKGROUND

Having worked with Clinical Data for over 20 years, I know of all of the struggles to setup, collect, report and submit a Clinical Study/Project to authorities. Keeping in control is a must and it is really hard work as there are so many things that you need to keep in your head. No single person has full control over what is put into the different sections of the eCTD, so we need standards, computers and links between documents to be able to navigate and digest the information. Unfortunately, we also need to repeat the same information in multiple places, as we want to present it multiple times in different contexts. So we copy e.g. the study design from the protocol to study reviewers guides and we put date of death in multiple domains, and consequently need to create strategies to make sure all documents/datasets are updated to contain the final version. Whatever we are doing, it is not DRY (Don't Repeat Yourself.)
That is why I'm excited that the pharma industry is moving in the direction of getting more precision in expressing data in machine understandable open formats, such as the Biomedical Concepts (BCs) and the CDISC Open Rules Engine (CORE.)
The idea for this presentation came after the last PHUSE EU Connect conference 2022 in Belfast, where I attended the CDISC CORE workshop.

INTRODUCTION - AN IDEA

As I have been working with BCs, I knew that they can be used for automation of most of the tasks involved in collecting and reporting clinical trials (e.g. SDTM annotation of collection forms, create SDTM datasets, define-xml etc.) At the conference last year, I gave a detailed explanation of how you can do this and showed that you don't not need new shiny toys, all you need for this is a web browser and a little knowledge of HTML and Javascript.
But there was one use case I hadn't looked at, using BCs to check data for conformance to standards, hence my interest in the CDISC CORE workshop, since a CORE rule should just be another perspective of the same data that a BC represents.
The workshop was great, and on the bus back from the conference in Belfast, the idea of creating CORE rules from BCs was born. Two hours later, when arriving at Dublin Airport the first set of BCs had been converted to CORE rules.

WHAT'S THE DIFFERENCE?

So what is the difference between a Core rule and Biomedical Concept? Here are some short definitions.

CORE RULES

Designed to check conformance, that the correct metadata is stored together on a row in database table, e.g. a dataset or a csv file.

BIOMEDICAL CONCEPTS

A BC has a broader use case. It should be:

- Identifiable - It has an identifier which is unique
- Complete - Everything is defined
- Atomic - If it is split, it loses meaning
- Data Specification - Specification of the data, not how it is used with a particular technology

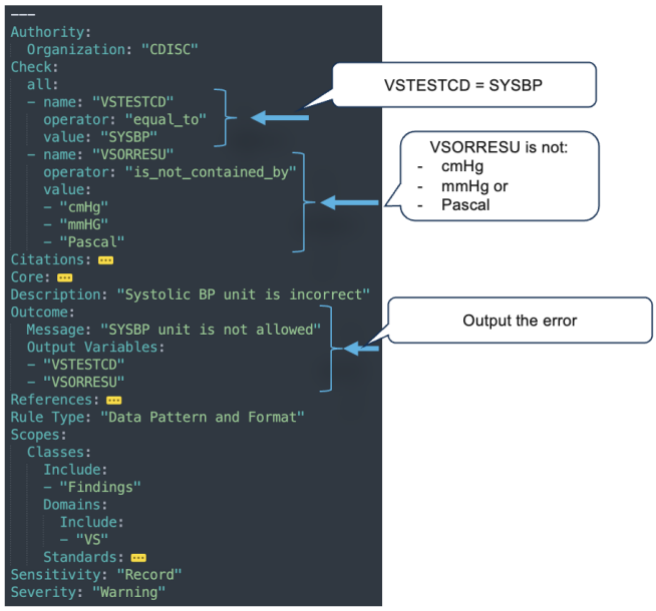
They represent different perspectives, or use cases, on the same data. The CORE rule has a more specific use case, whereas the BC is general in that it is defining the complete set of information that a specific concept contains.

PROOF OF CONCEPT

For this presentation, it is the second and fourth bullets of the BC that are the important ones.
So if a BC is complete and a data specification, the CORE rule should be in the BC. Which means that it should be possible to generate the CORE rule from the information in the BC.
The content of them is explained below, but in abbreviated form to show the necessary information for comparison.
The raw full content of the files is included in appendix 1 and 2 at the end of the document.

THE INSIDE OF A CORE RULE

The below picture shows a CORE rule (in YAML format) generated from the BC for Systolic Blood Pressure. It checks whether a correct unit has been used when the test code in the Vital Signs domain is SYSBP.
N.B. some sections in the file has been collapsed to emphasize the important sections.



CORE rule details

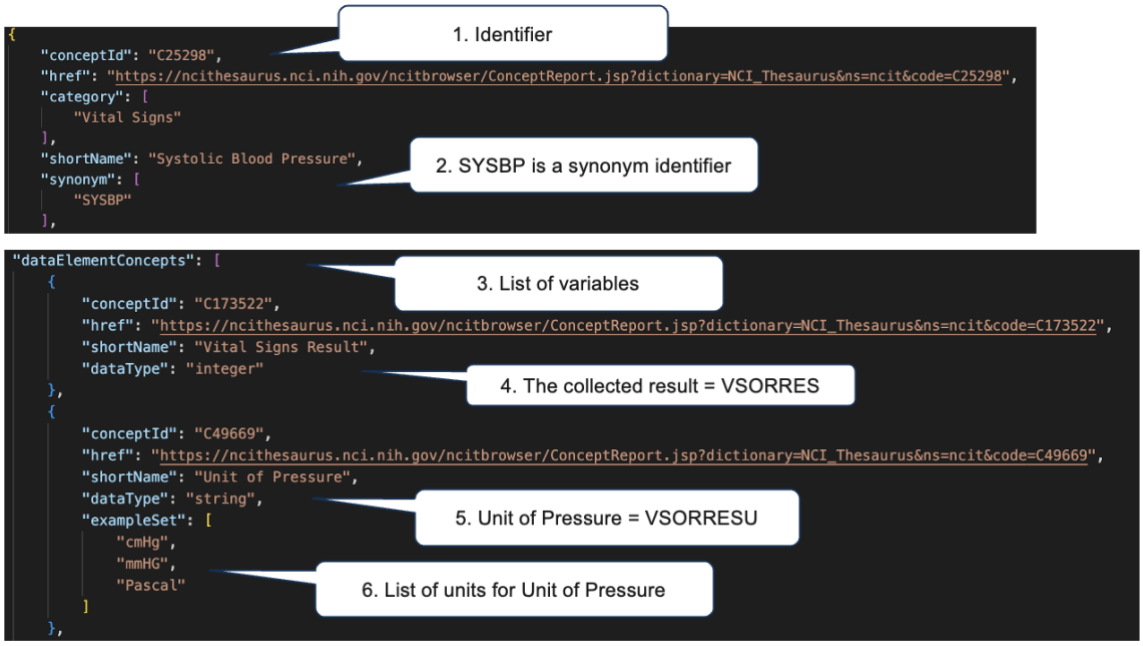
As indicated by the arrows above, this check contains:

1. a specification of which test code is applicable for performing the check
2. a specification of allowed units for VSORRESU
3. the message to output if VSORRESU contains a value not specified in 2

THE INSIDE OF A BC

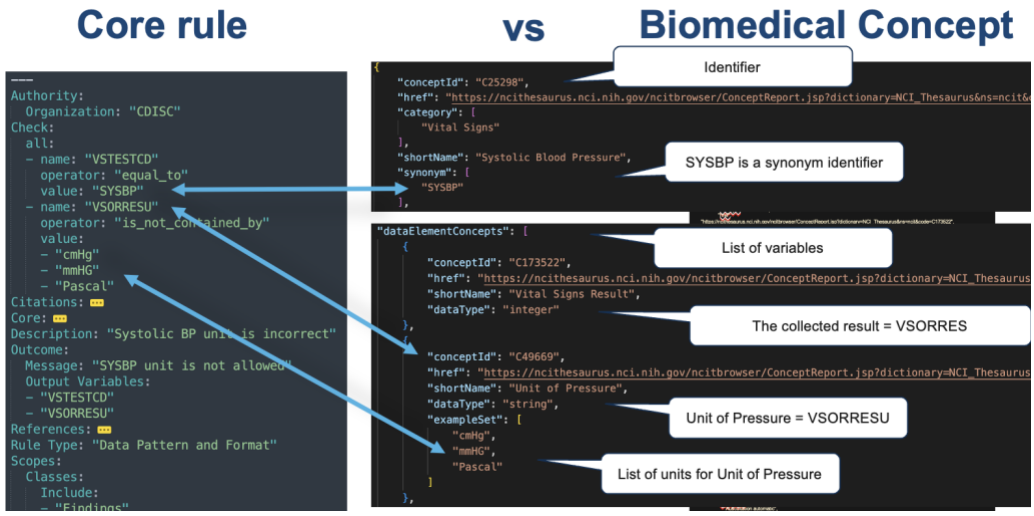
As the BC is complete and contains more details (e.g. terminology for position and location) it has been abbreviated more to show the source of the generated CORE rule above.
As indicated by the arrows in the picture below, the BC contains:

1. an identifier of the concept
2. the synonym identifier to use as a test code in a domain
3. the list of data elements (variables) related to the concept
4. the data element for the collected result
5. the data element for the unit of the collected result
6. the list of units applicable for the concept



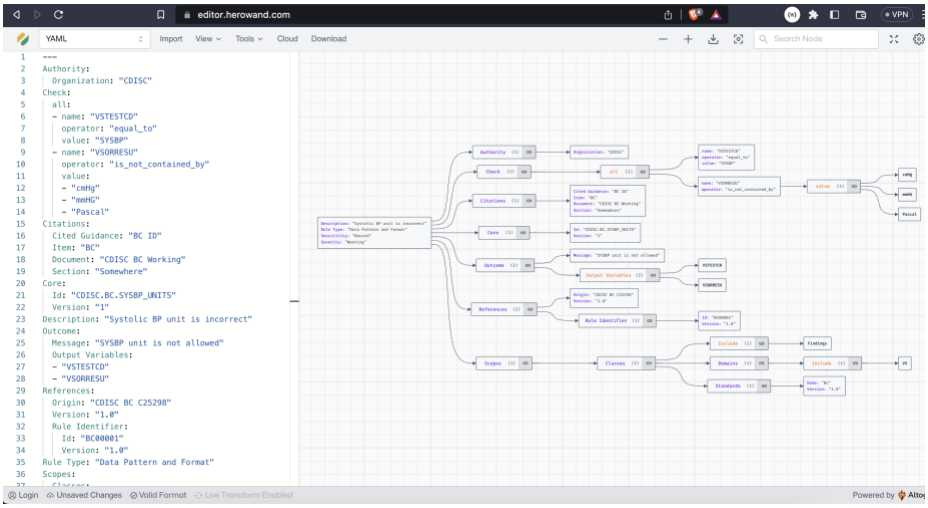
Biomedical Concept details

If we put them side-by-side, you can see how the information has been copied from the BC.

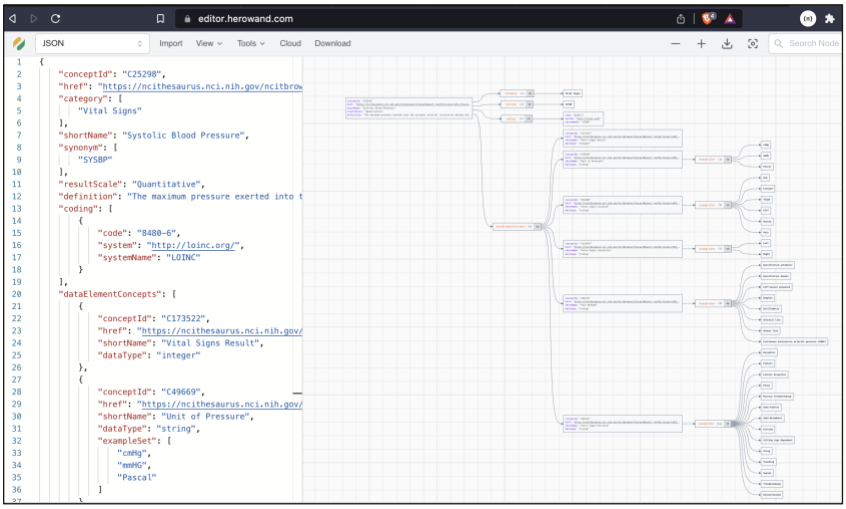


CORE rule vs Biomedical Concept

The good thing with YAML and JSON formats is that we can also view them graphically, e.g. with JSONcrack. Just paste any YAML or JSON file and view the results.

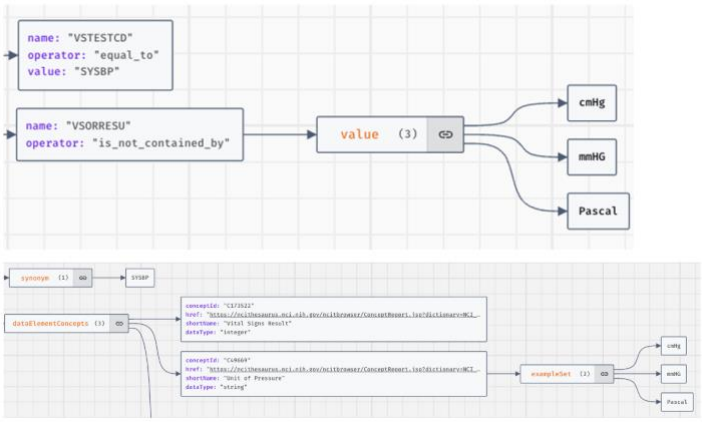


CORE rule viewed in JSONCrack/Herowand



BC viewed in JSONCrack/Herowand

Visualizing the files in this way show even more clearly how similar the representation of this information is:



The common details from the CORE rule and BC

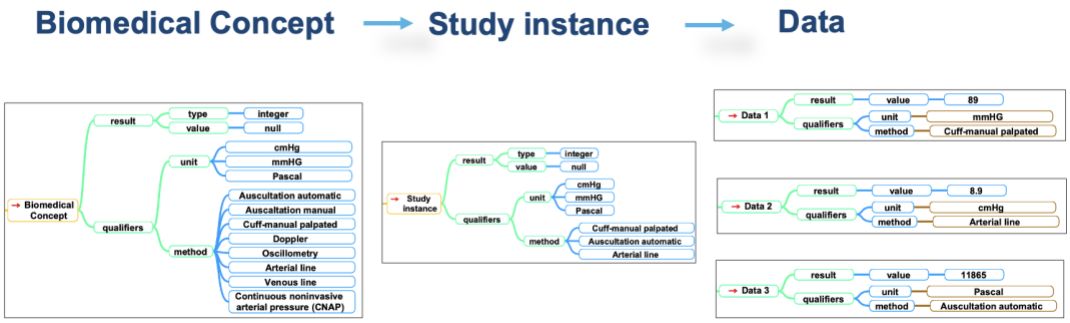
The CORE rule created from the BC successfully ran in the CORE engine after creating some test data.
N.B. the test data is normal SDTM datasets, not YAML or JSON formatted files. So the CORE rule can be applied directly to existing SDTM datasets.

As the BC contains all qualifiers and associated terminology, all CORE rules can be created from it, not only SYSBP as showed above in the proof of concept.

It should even be possible to trigger the CORE engine from the BC directly, so there shouldn't be a need to create separate artefacts for validation. This would make our data flow more DRY as it is just about different perspectives on the same data, viewed from different use cases.

BCs IN THE DATA FLOW

As shown in my presentation last year, the BC can also be viewed as a form, and we can create a study specific instances of them to represent the needs of an individual study. The data in the study can be collected via the forms generated and the resulting data can be checked directly whether it conforms to the BC instances setup for that study.

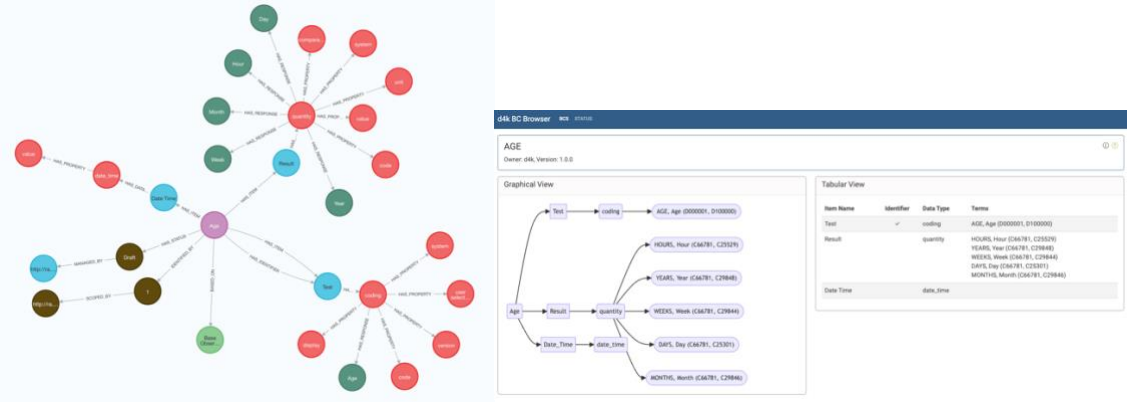


Data flow

The BC is the source of truth for the Study instance. The Study instance is the source of truth for the data. As they are all linked and referencing back to the previous artefact, the data only need to carry the collected results as it links directly back to the study instance where any pre-set information is available. So the information for define-xml is also available in the BC.

The data above has been depicted as a visualisation of a JSON/YAML, but as stated earlier, the CORE rule works on existing SDTM dataset, and we can use other approaches than sending cumulative "rectangular" files which is most common today. An API could send JSON responses and validation could be performed on the response.

In the ongoing Phase Three of TransCelerate Digital Data Flow (DDF) the current work is focusing on integrating the CDISC USDM with ICH M11 (ICH Clinical Electronic Structured Harmonised Protocol (CeSHarP)) and attaching BCs to timepoints and encounters in the USDM graph.



BC in native graph format, visualised as a tree layout and a tabular view

When we have protocols in USDM, attaching BCs to the USDM schedule of events, the paper protocol, eCRF, SDTM datasets and validation checks are only different queries and serializations from the same graph.

CONCLUSION

We are moving to a data centric world where linked data and graph databases play a key role, as they provide the means to express our complex clinical data in more detail and explicitly link information, so it does not need to be repeated. Initiatives like CORE and BCs will provide new ways of working and enable automation of many of today's manual task, so that we can improve the clinical data flow, get better quality data and be in control.

REFERENCES

CDISC CORE https://www.cdisc.org/core
CDISC Biomedical Concepts https://www.cdisc.org/cdisc-biomedical-concepts
Transcelerate DDF https://www.transceleratebiopharmainc.com/initiatives/digital-data-flow/
DRY (Don't repeat yourself) https://en.wikipedia.org/wiki/Don%27t_repeat_yourself
JSONcrack https://jsoncrack.com/editor
CDISC DDF USDM Package https://pypi.org/project/usdm/
Transcelerate DDF https://www.transceleratebiopharmainc.com/initiatives/digital-data-flow/
JSON https://www.json.org/json-en.html
YAML https://yaml.org/

My paper, presentation and code from PHUSE EU Connect 2022 in Belfast:
SM01: Release Your Creativity on Your Company Laptop:
Paper: https://phuse.s3.eu-central-1.amazonaws.com/Archive/2022/Connect/EU/Belfast/PAP_SM01.pdf
Presentation: https://phuse.s3.eu-central-1.amazonaws.com/Archive/2022/Connect/EU/Belfast/PRE_SM01.pdf
The github repository with example code: https://github.com/jozza/simple-data-template

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:
Johannes Ulander
data4knowledge ApS
Email: ju@data4knowledge.dk
Web: https://d4k.dk/

APPENDIX 1 – FULL CORE RULE FOR SYSBP (YAML FORMAT)

```
---
Authority:
  Organization: "CDISC"
Check:
  all:
  - name: "VSTESTCD"
    operator: "equal_to"
    value: "SYSBP"
  - name: "VSORRESU"
    operator: "is_not_contained_by"
    value:
    - "cmHg"
    - "mmHG"
    - "Pascal"
Citations:
  Cited Guidance: "BC ID"
  Item: "BC"
  Document: "CDISC BC Working"
  Section: "Somewhere"
Core:
  Id: "CDISC.BC.SYSBP_UNITS"
  Version: "1"
Description: "Systolic BP unit is incorrect"
Outcome:
  Message: "SYSBP unit is not allowed"
  Output Variables:
  - "VSTESTCD"
  - "VSORRESU"
References:
  Origin: "CDISC BC C25298"
  Version: "1.0"
  Rule Identifier:
    Id: "BC00001"
    Version: "1.0"
Rule Type: "Data Pattern and Format"
Scopes:
  Classes:
    Include:
    - "Findings"
    Domains:
      Include:
      - "VS"
    Standards:
      Name: "BC"
      Version: "1.0"
Sensitivity: "Record"
Severity: "Warning"
```

APPENDIX 2 – FULL BC FOR SYSBP (JSON FORMAT)


```
{
    "_links": {
        "parentBiomedicalConcept": {
            "href": "/mdr/bc/packages/2022-10-26/biomedicalconcepts/C54706",
            "title": "Blood Pressure",
            "type": "Biomedical Concept"
        },
        "parentPackage": {
            "href": "/mdr/bc/packages/2022-10-26/biomedicalconcepts",
            "title": "Biomedical Concept Package Effective 2022-10-26",
            "type": "Biomedical Concept Package"
        },
        "self": {
            "href": "/mdr/bc/packages/2022-10-26/biomedicalconcepts/C25298",
            "title": "Systolic Blood Pressure",
            "type": "Biomedical Concept"
        }
    },
    "conceptId": "C25298",
    "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C25298",
    "category": [
        "Vital Signs"
    ],
    "shortName": "Systolic Blood Pressure",
```

```
    "synonym": [
        "SYSBP"
    ],
    "resultScale": "Quantitative",
    "definition": "The maximum pressure exerted into the systemic arterial circulation during the contraction of the left ventricle of the
heart.",
    "coding": [
        {
            "code": "8480-6",
            "system": "http://loinc.org/",
            "systemName": "LOINC"
        }
    ],
    "dataElementConcepts": [
        {
            "conceptId": "C173522",
            "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C173522",
            "shortName": "Vital Signs Result",
            "dataType": "integer"
        },
        {
            "conceptId": "C49669",
            "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C49669",
            "shortName": "Unit of Pressure",
            "dataType": "string",
            "exampleSet": [
                "cmHg",
                "mmHG",
                "Pascal"
            ]
        },
        {
            "conceptId": "C83088",
            "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C83088",
            "shortName": "Vital Signs Location",
            "dataType": "string",
            "exampleSet": [
                "Arm",
                "Forearm",
                "Thigh",
                "Calf",
                "Artery",
                "Vein"
            ]
        },
        {
            "conceptId": "C123975",
            "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C123975",
            "shortName": "Vital Signs Laterality",
            "dataType": "string",
            "exampleSet": [
                "Left",
                "Right"
            ]
        },
        {
            "conceptId": "C82535",
            "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C82535",
            "shortName": "Test Method",
            "dataType": "string",
            "exampleSet": [
                "Auscultation automatic",
                "Auscaltation manual",
                "Cuff-manual palpated",
                "Doppler",
                "Oscillometry",
                "Arterial line",
                "Venous line",
                "Continuous noninvasive arterial pressure (CNAP)"
            ]
        },
        {
            "conceptId": "C83114",
            "href": "https://ncithesaurus.nci.nih.gov/ncitbrowser/ConceptReport.jsp?dictionary=NCI_Thesaurus&ns=ncit&code=C83114",
            "shortName": "Vital Signs Position",
            "dataType": "string",
            "exampleSet": [
                "Decupitus",
                "Fowlers",
                "Lateral Decupitus",
                "Prone",
                "Reverse Trendelenburg",
                "Semi-Fowlers",
                "Semi-Recumbent",
                "Sitting",
                "Sitting Legs Dependent",
                "Sling",
                "Standing",
                "Supine",
                "Trendelenburg",
                "Unconstrained"
            ]
        }
    ]
}
```