The iCRF Generator
- Generate interoperable CRFs -

Sander de Ridder
Jeroen Beliën
Today

- Use case
- Original features
- New features NWO-funded version
- Demo
When designing a new electronic CRF...
When designing a new electronic CRF...

In my new study my crf will look like this:

Please select all applicable symptoms at first presentation

- Has the patient received chemotherapy for prior malignancy?
  - Yes
  - No
When designing a new electronic CRF...

I will use:
0 = Yes
1 = No

Researcher A
When designing a new electronic CRF...

I will use:
0 = Yes
1 = No

Researcher A
Researcher B
When designing a new electronic CRF...

I will use:
0 = Yes
1 = No

In my new study my crf will look like this:
When designing a new electronic CRF...

Researcher A

I will use:
0 = Yes
1 = No

Researcher B

I will use:
0 = No
1 = Yes
When designing a new electronic CRF...

I will use:
0 = Yes
1 = No

I will use:
0 = No
1 = Yes

Let’s combine the data!
When designing a new electronic CRF...

I will use:
0 = Yes
1 = No

I will use:
0 = No
1 = Yes

Let’s combine the data!

It’s a mess!
Data not interoperable!
Let’s try again, but a bit more interoperable...
Let’s try again, but a bit more interoperable...

In my study I will reuse definitions from the Basic Health Data Set!

Researcher A
Let’s try again, but a bit more interoperable...

In my study I will reuse definitions from the Basic Health Data Set!

Me too!
Let’s try again, but a bit more interoperable...

In my study I will reuse definitions from the Basic Health Data Set!

Me too!

Let’s combine the data!
Let’s try again, but a bit more interoperable...

In my study I will reuse definitions from the Basic Health Data Set!

Me too!

Much easier to combine! Interoperable!

Let’s combine the data!
Let’s try again, but a bit more interoperable…

In my study I will reuse definitions from the Basic Health Data Set!

Me too!

Let’s combine the data!

Much easier to combine! Interoperable!

Bonus: a happy PI
Let’s try again, but a bit more interoperable…

In my study I will reuse definitions from the Basic Health Data Set!

Me too!

Let’s combine the data!

Much easier to combine! Interoperable!

Reuse of such definitions should be easy!

Bonus: a happy PI
The iCRF Generator

• Easy to use tool to generate interoperable Case Report Forms
  • Uses codebooks stored in ART-DECOR, e.g.
    • Basic Health Data Set (Basisgegevensset Zorg)
    • Clinical Building Blocks (Zorginformatiebouwstenen)
    • RIVM population screening
    • 2 PALGA Protocols
    • VASCA
    • FAIR Genomes
  • Supports
    • OpenClinica 3
    • Castor (Step, Report, Survey)
    • REDCap
    • Molgenis EMX
iCRF Generator Extended Edition

Opening Interoperable Science

• Easy to use tool to generate Interoperable Case Report Forms
  • Support for alternative codebook sources
    • ART-DECOR
    • OpenEHR (open standard specification)
      • Clinical Knowledge Manager (https://ckm.openehr.org/)
  • Supports
    • OpenClinica 3
    • Castor (Step, Report, Survey)
    • REDCap
    • Molgenis EMX
    • ODM-XML (CDISC)
  • Improved user interface for codebook selection
Current iCRF Generator Flow

Illustration of the current general flow inside the application.
Demo

Hip arthroplasty component:

https://ckm.openehr.org/ckm/archetypes/1013.1.1748
Codebooks are made available in ART-DECOR or openEHR, e.g.
• Basic Health Data Set (Basisgegevensset Zorg)
• RIVM population screening
• VASCA
→ Reuse these definition in your CRFs to improve interoperability of data!

The iCRF Generator
• Allows user to reuse these codebooks thereby generating interoperable CRFs (iCRFs)
• Supports multiple EDCs
→ Makes it easier to create interoperable CRFs!

(Inter)nationally accepted codebooks
Codebook Sources

etc…

castor
REDCap
OpenClinica
To schedule a demo or learn more

• Contact us:
  • Sander de Ridder - a.deridder1@amsterdamumc.nl
  • Jeroen Beliën - jam.belien@amsterdamumc.nl

• Read the paper:
  • https://f1000research.com/articles/9-81

• Download the tool and give it a try (NWO-version not yet released):
  • Tool - https://github.com/aderidder/iCRFGenerator/releases
  • Source code - https://github.com/aderidder/iCRFGenerator
1. Select the EDC of your choice
2. Click Run
The iCRF Generator - Usage

3. Select one or more codebooks
4. Click next

5. Select version(s) and language(s)
6. Click next
7. Select the items and codes for your CRF
8. Click next
9. Finish the Wizard and Save your CRF
10. After saving, import the CRF into your EDC of choice or edit the CRF with a suitable editor