

The BREX – beauty or beast?



S1000D User Forum, Montreal, 2011-06-06--08

Svante Ericsson

Agenda

- What is the BREX?
- The history - why was the BREX invented and introduced?
- What does it look like?
- How does it fit into the publication process?
- How to utilize it?



What is the BREX?

- It is a *data module*, like any other data module (almost)
- It contains *rules* to which the *objects* in a CSDB must adhere
- It can contain *rules* to control *management* of a CSDB
- It constitutes a *formalized* way to document and exchange *business rules* (ie rules for implementation of S1000D)
- In principle, the BREX is *needed to understand* all other objects

The BREX history in brief

- Born in Wyboston 2003
- Introduced in Issue 2.2
- Considerable lift in Issue 4.0
 - Default BREX more comprehensive
- Form and function fairly stable since start

Why was it introduced?

- The first task of the BREX was to specify how project configurable attributes are used by a project
 - *Solve the potential presentation conflict!*

Our business case ...

On one hand, suppose there are two vendors:

- They both produce their specific Materiel
- They both provide technical publications to their Materiel
- They both have differing views of what is a good publication
- They both have developed their own terminology
- They both have developed their own IETP viewers

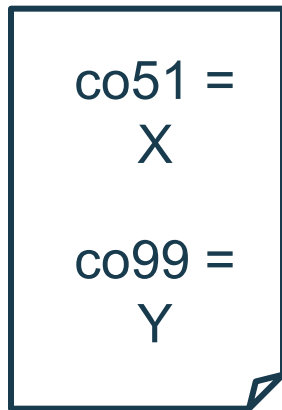
Our business case ...

On the other hand, suppose there is one Materiel user:

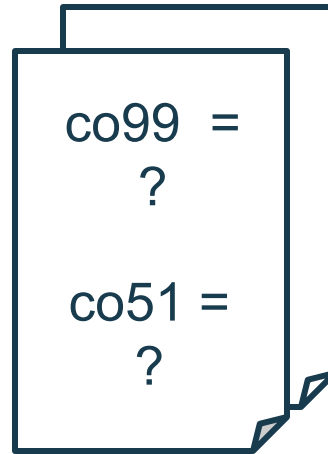
- He has acquired all [...] vendors' systems and integrated them
- His integrated Materiel calls for integrated publications
- Indeed, his terminology is by far the best one
- He doesn't like the vendors' data viewers - his own is much better

The confusion

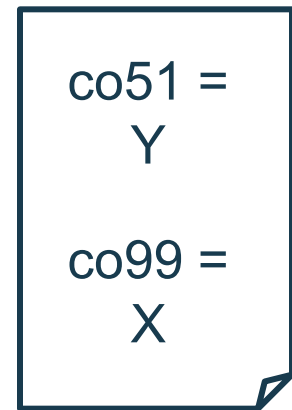
1st vendor



The integrator

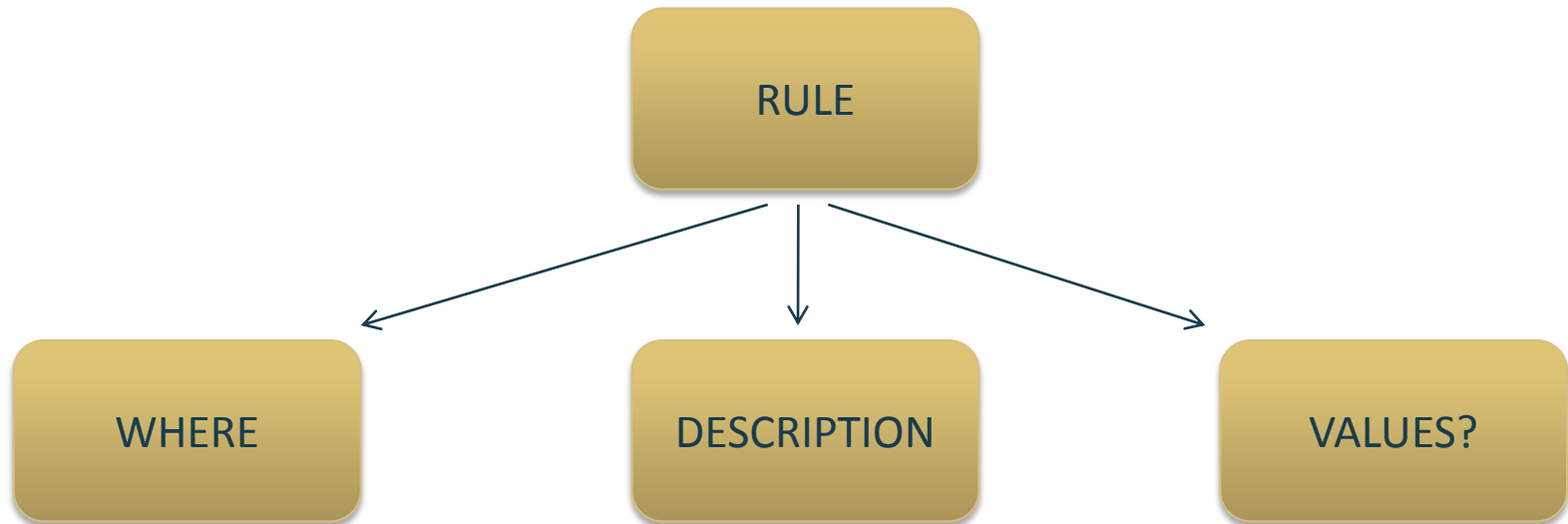


2nd vendor



- With issue 4.1, every xml CSDB object requires reference to the governing BREX
- Three types of rules
 - SNS rules – specify SNS applied
 - Context related rules – per schema or for all schemas (elements, attributes, media/graphics formats)
 - Non-context rules – not tied to schema structures, etc
- Currently, restricted to rules that apply within one CSDB object only, i.e. no combinations of objects!
- BREX data modules can be layered

The context related rules



Rule target:

- Computer readable
- Based on XPath
- **Computer geek**

The narrative rule:

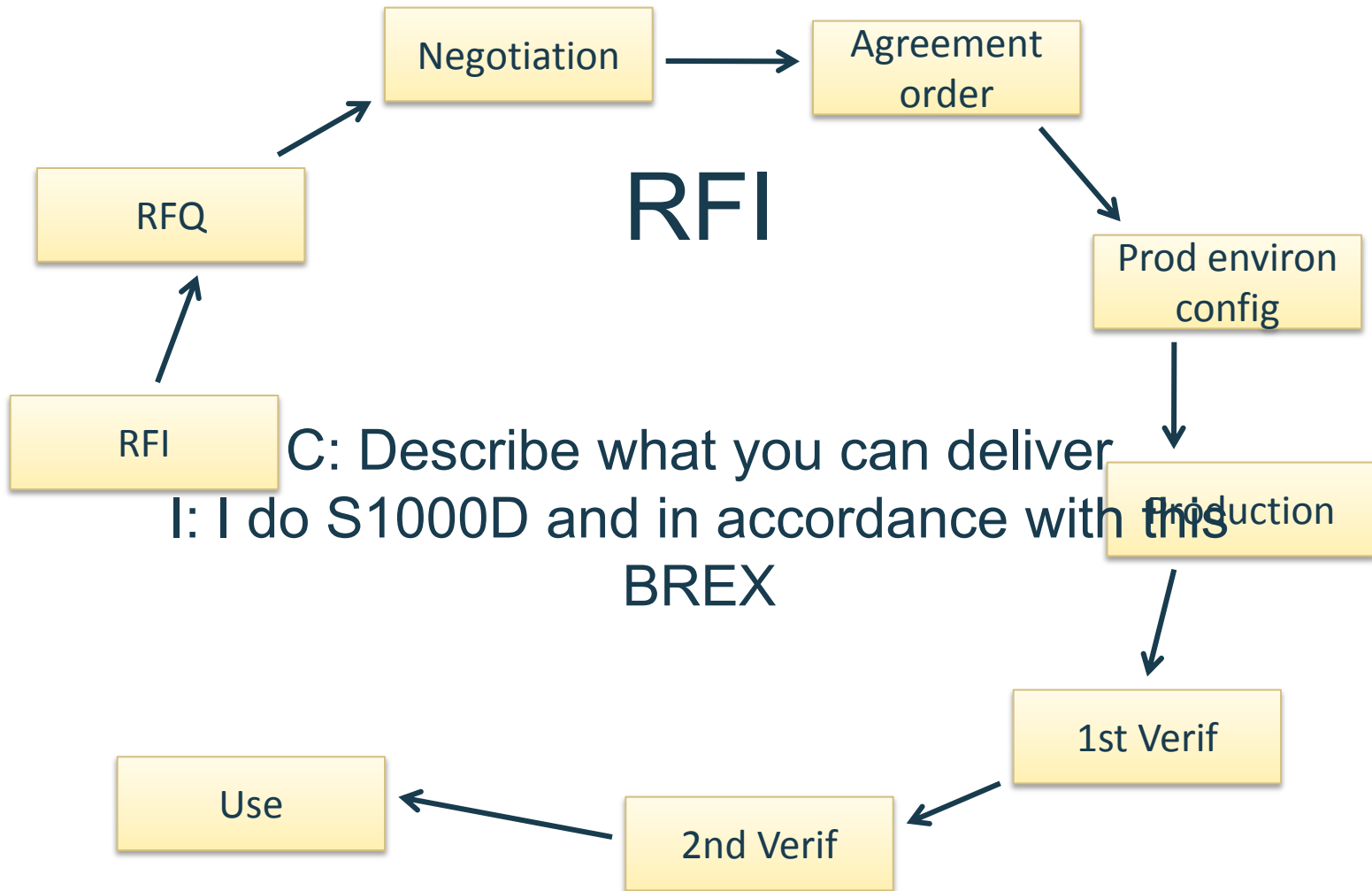
- Human readable
- Refers to source
- **Publ manager**

Allowable values:

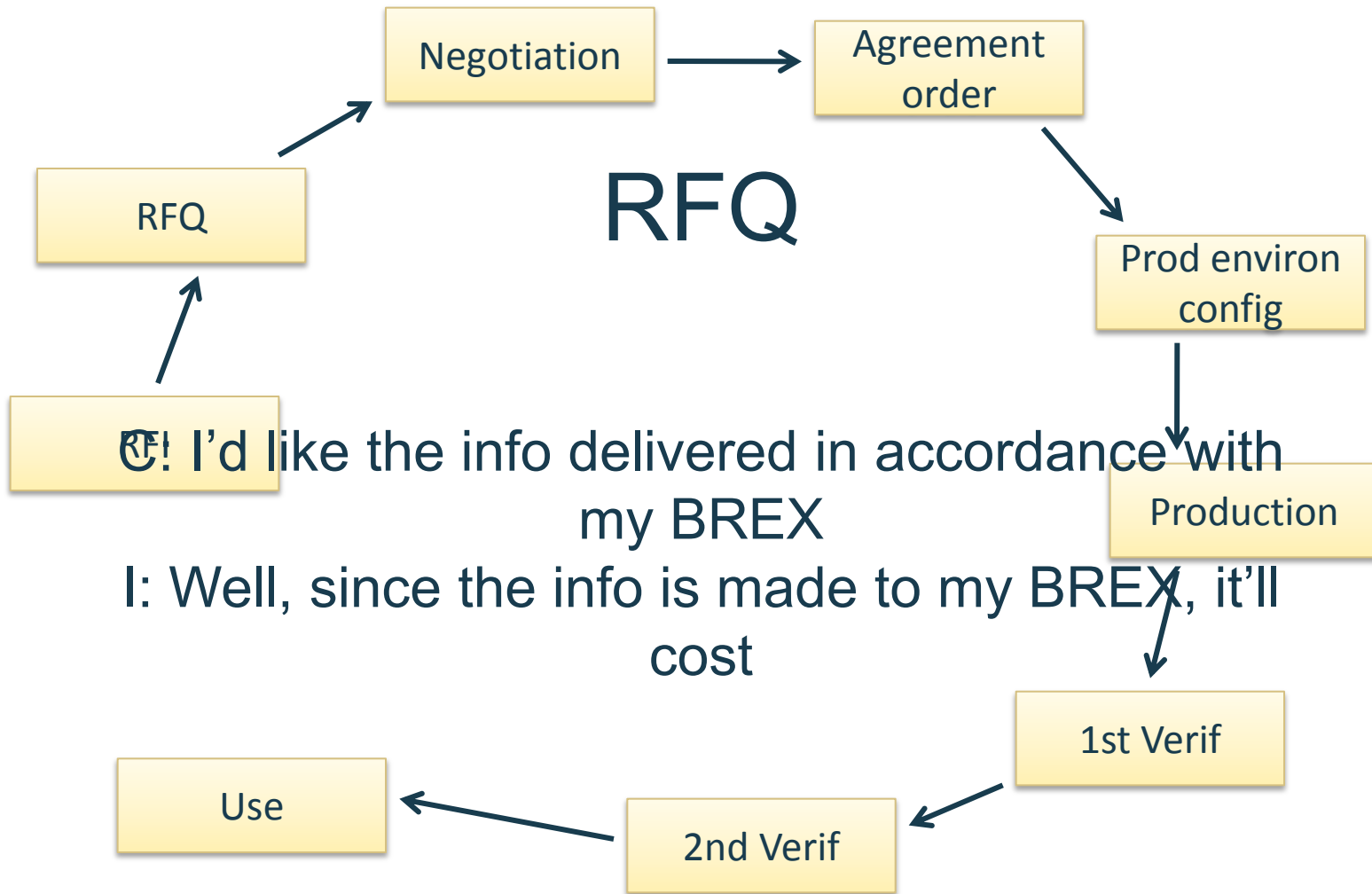
- Humans/computers
- Only as needed
- **Manager & geek**

- Comes with every issue since 2.2
- Contains rules not captured by xml schemas
- Issue 2.2
 - Approx 15 rules
 - All rules except one defined sets of attribute values
 - No narrative rules
- Issue 4.1
 - Approx 150 rules
 - Ca. 70 rules define sets of item values (attributes)
 - Ca. 25 rules are purely narrative

BREX in the acquisition/production process



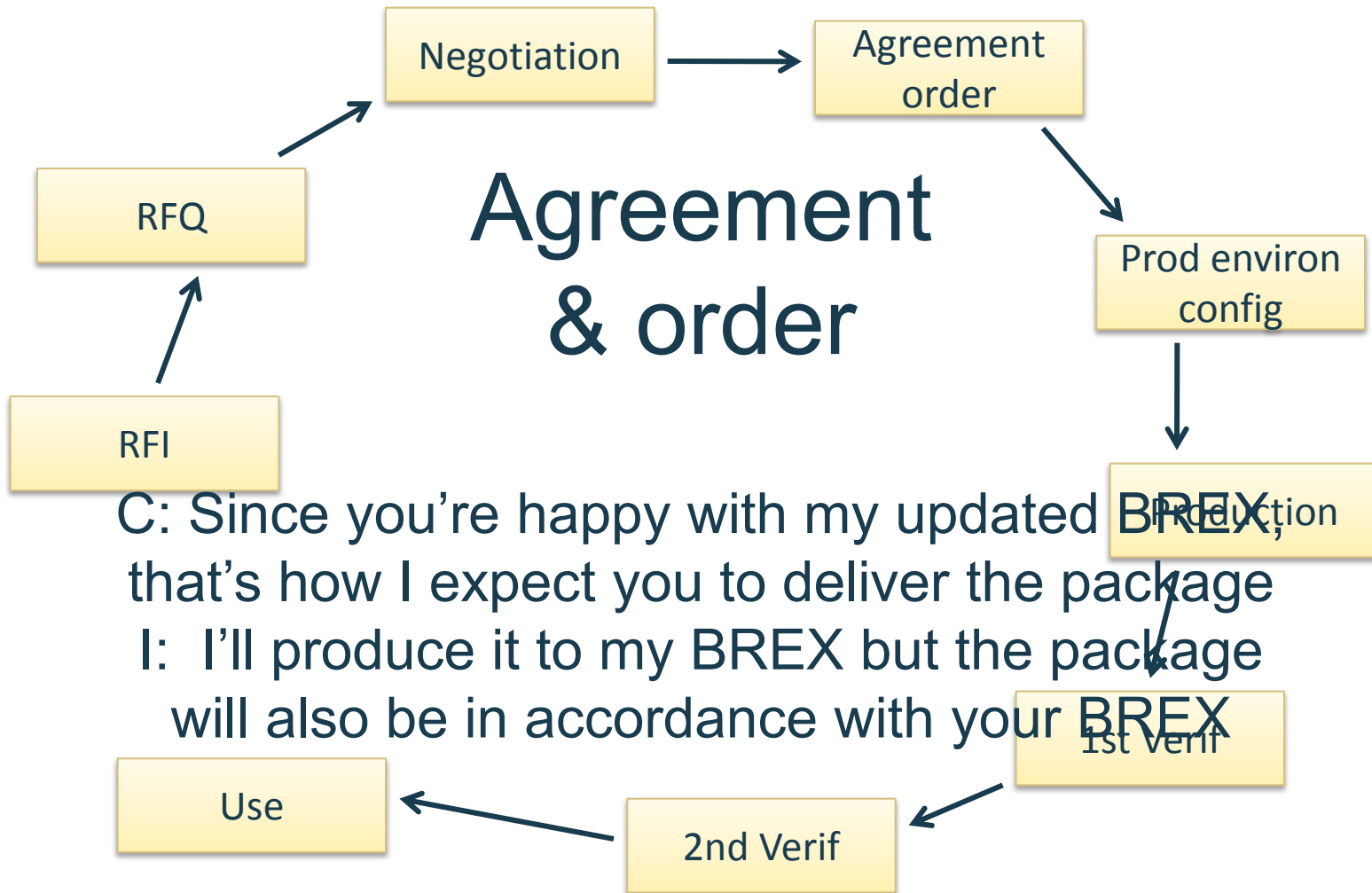
BREX in the acquisition/production process



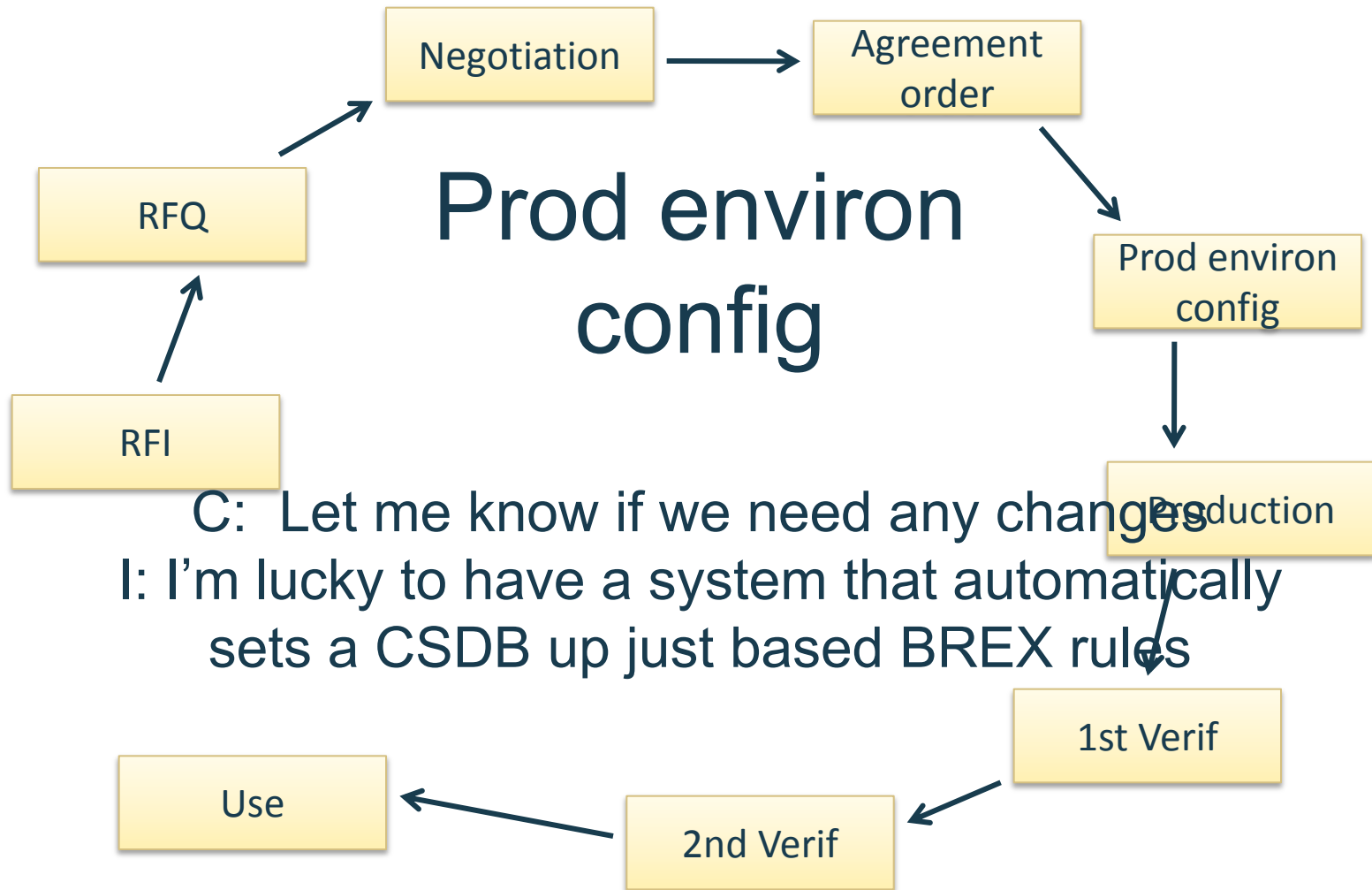
BREX in the acquisition/production process



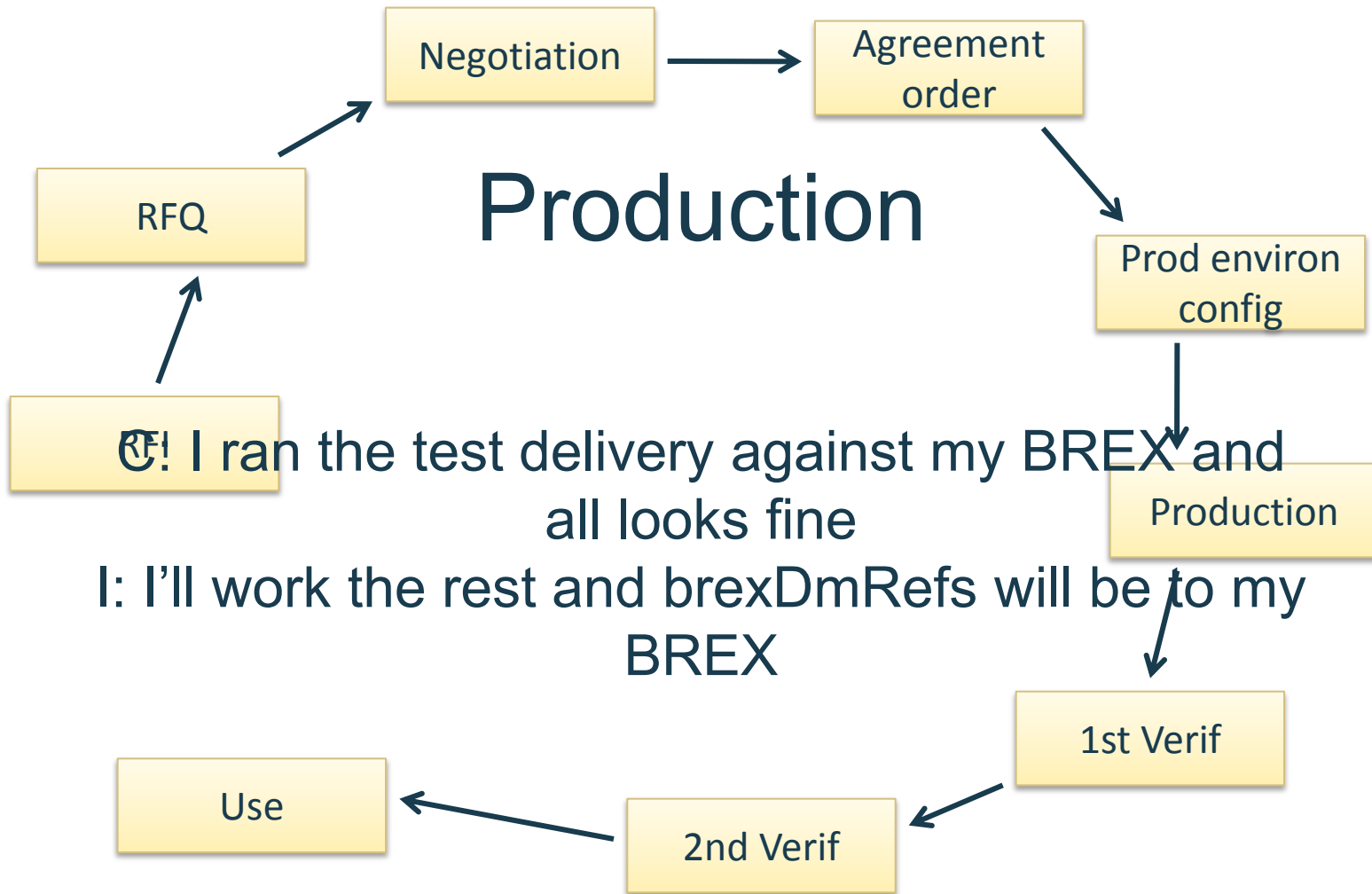
BREX in the acquisition/production process



BREX in the acquisition/production process



BREX in the acquisition/production process



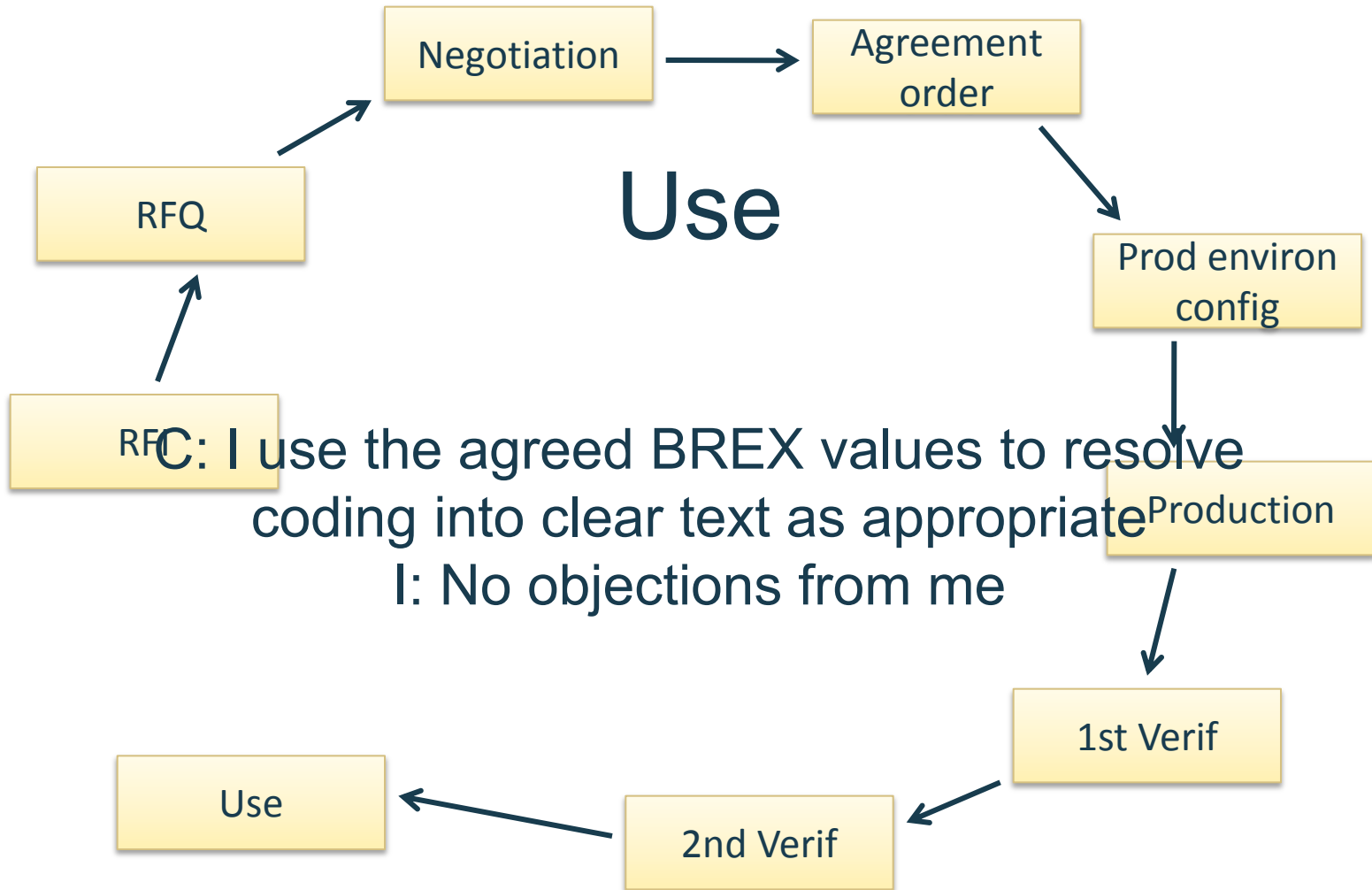
BREX in the acquisition/production process



BREX in the acquisition/production process



BREX in the acquisition/production process



How can the BREX be used?

- The BREX data module can be used for several purposes:
 1. In developing BRs for a project, ie for exchanging BRs
 2. To inform about your S1000D view
 3. To configure a system to the rules of a BREX module
 4. To reflect a system configuration (export to a BREX)
 5. To verify data modules in production
 6. To verify a delivered batch of data modules
 7. To interpret codes for presentation purposes
 8. ...

So, beauty or beast?



Beautiful Beast!

Thanks for your attention!

Questions?

svante.ericsson@corena.com