

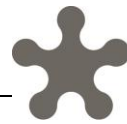


Leidraad VISI-systematiek versie 1.6

Bijlage 7 Richtlijn voor 'Successor'

Normatief

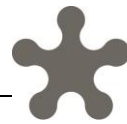
Documentversie: 1.1
Datum: april 2019
Status: Definitief



VISI 2003 - 2019

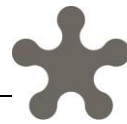
Op deze uitgave is de Creative Commons Licentie – Naamsvermelding – NietCommercieel – GelijkDelen – van toepassing. (zie: <http://creativecommons.org/licenses/by-nc-sa/3.0/nl/>)

CROW en degenen die aan deze publicatie hebben meegewerkt, hebben de hierin opgenomen gegevens zorgvuldig verzameld naar de laatste stand van wetenschap en techniek. Desondanks kunnen er onjuistheden in deze publicatie voorkomen. Gebruikers aanvaarden het risico daarvan. CROW sluit, mede ten behoeve van degenen die aan deze publicatie hebben meegewerkt, iedere aansprakelijkheid uit voor schade die mocht voortvloeien uit het gebruik van de gegevens.



Inhoud

1	Rules for 'Successor'	4
1.1	<i>A successor has always the SAME ROLE</i>	4
1.2	<i>A successor can NEVER be changed</i>	4
1.3	<i>NO LOOP of successors</i>	4
1.4	<i>A successor of a successor is allowed</i>	4
1.5	<i>A successor of several persons (in role)</i>	4
1.6	<i>A predecessor can NOT start a transaction</i>	4
1.7	<i>A PersonInRole with a successor can NOT send a message</i>	4
1.8	<i>An initiator and executor of a transaction will NEVER change</i>	5
2	Examples	5
2.1	<i>Example 1</i>	5
2.2	<i>Example 2</i>	6
2.3	<i>Example 3</i>	6



1 Rules for 'Successor'

For a correct interpretation of all the possibilities of a successor certain rules need to be followed. First the do's and don't's are presented, followed by some examples.

1.1 *A successor has always the SAME ROLE*

If 'B' is a successor of 'A' then 'B' must have the SAME role as 'A'.

1.2 *A successor can NEVER be changed*

If 'A' has a successor 'B' then 'B' will ALWAYS be the successor of 'A'. Later on 'C' cannot be the successor of 'A', but can become the successor of 'B'.

1.3 *NO LOOP of successors*

If 'B' is a successor of 'A' then 'A' cannot be a successor of 'B'.

1.4 *A successor of a successor is allowed*

At first 'B' is a successor of 'A'. But after some time 'B' can also have a successor. Theoretically such a chain of successors can be unlimited, but it can never become a loop.

Keep in mind that following situation can occur:

- (1) 'A' starts transaction T₁;
- (2) 'B' becomes a successor of 'A';
- (3) 'B' replies and sends a messages in transaction T₁;
- (4) 'B' starts transaction T₂;
- (5) 'C' becomes a successor of 'B';
- (6) 'C' replies and sends messages in transaction T₁ and T₂.

Maybe later on 'D' will become a successor of 'C' then 'D' will be responsible for T₁ and T₂ (if T₁ and T₂ are not finished). 'A' cannot be the successor of 'C'.

1.5 *A successor of several persons (in role)*

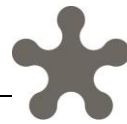
'B' can become the successor of several persons (in role). In this case 'B' will be responsible for all open transactions of all predecessors.

1.6 *A predecessor can NOT start a transaction*

A PersonInRole with a successor (=predecessor) is not an active member of the project and therefore cannot start a new transaction.

1.7 *A PersonInRole with a successor can NOT send a message*

A PersonInRole with a successor (=predecessor) is not an active member of the project and therefore cannot send a message.



1.8 An initiator and executor of a transaction will NEVER change

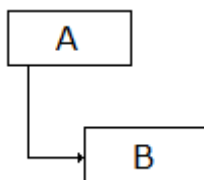
For example 'A' starts a new transaction T1 and sends a message to 'B'. The VISI xml-message will contain 'A' as the initiator and 'B' as the executor.

When 'C' becomes the successor of 'B' and replies on behalf of 'B'. The VISI xml-message will contain 'A' as the initiator and 'B' as the executor. 'C' is also included, but only as the successor of 'B'.

2 Examples

2.1 Example 1

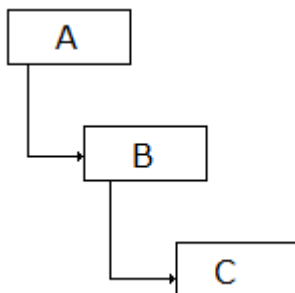
Most simple case is when 'B' is a successor of 'A'. It can be displayed like this:



The following notation is used to show that 'B' is a successor of 'A':

$A \rightarrow B$

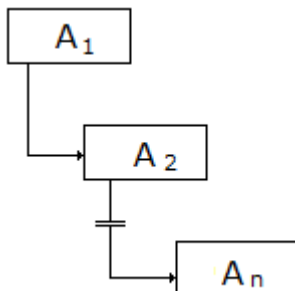
'B' also can have a successor and the diagram will look like this:



This can be written like:

$A \rightarrow B \rightarrow C$

In common case the "successor-predecessor" diagram can be displayed like:

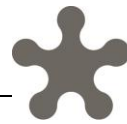


Or:

$A_1 \rightarrow A_2 \rightarrow \dots \rightarrow A_n$

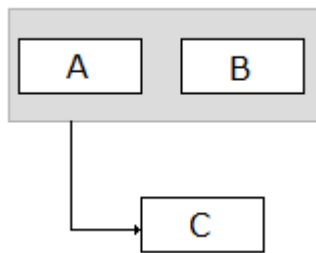
Where:

$A_1 \neq A_2 \neq \dots \neq A_n$



2.2 Example 2

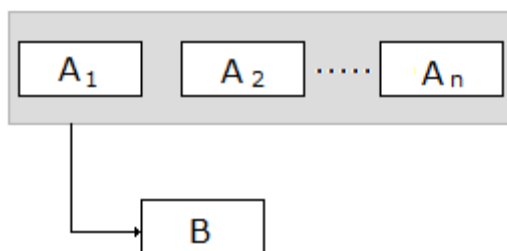
'C' can become the successor of several persons (in role). In a diagram:



Or:

$(A; B) \rightarrow C$

In common case the "successor-predecessor" diagram can be displayed like:



Or:

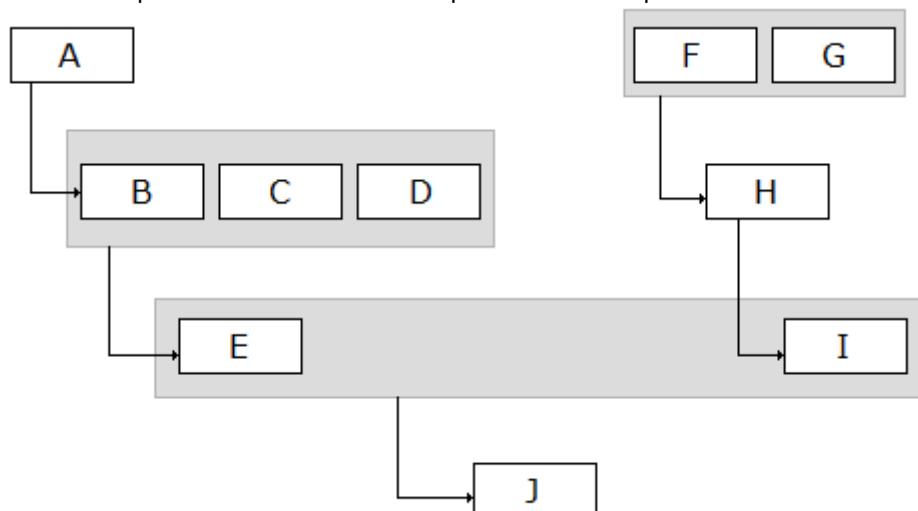
$(A_1; A_2; \dots; A_n) \rightarrow B$

Where:

$A_1 \neq A_2 \neq \dots \neq A_n$

2.3 Example 3

A final example which shows how complex "successor-predecessor" relations can be:



Or:

$((A \rightarrow B; C; D) \rightarrow E; (F; G) \rightarrow H \rightarrow I) \rightarrow J$

In this example 'J' is responsible for all open transaction of 'A', 'B', ..., 'I'.

< einde Bijlage 7 >