PROGETTI IV CALL 2023/2024

Titolo progetto:	**Dal questionario elettronico al piano di compatibilità – Get the Rules** *A cura di Simona Rosati. I paragrafi "Descrizione", "Obiettivi", e "Metodologia-Scenario 1" sono stati redatti da Adele Maria Bianco; i paragrafi "Scenario 2", "Conclusioni" e "Risultati Ottenuti" sono stati redatti da Laura Tosco. Sviluppo del traduttore Java a cura di Luigi Arlotta; finalizzazione del traduttore Java in regole R a cura di Adele Maria Bianco e Laura Tosco.*
Descrizione	Nell'ambito di un processo di controllo e correzione la fase di definizione delle regole di compatibilità spesso richiede un notevole impiego di risorse umane e di tempo. Ciò è particolarmente vero nel contesto delle rilevazioni statistiche strutturali (inclusi i censimenti), che comprendono un numero molto elevato di variabili rilevate. L'idea originaria del progetto nasce dall'osservare che nelle indagini CAI (*Computer Assisted Interviewing*) il processo di sviluppo del questionario elettronico comporta necessariamente che le regole di compilazione del questionario stesso siano implementate in procedure informatiche, secondo uno specifico *software* o linguaggio di programmazione (es. C++, Java, etc.) o attraverso sistemi di acquisizione dati generalizzati (es. Gino ++, Panda, Blaise etc.). Allo stesso modo, chi ha il compito di realizzare il piano di compatibilità deve trasformare le medesime regole del questionario in regole di controllo, secondo la sintassi del *software* adottato. Ciò significa che, sia l'analista informatico, sia l'esperto di controllo e correzione, entrambi sono tenuti a svolgere lo stesso processo mentale. Introducendo una fase di traduzione automatica delle regole di compatibilità si otterrebbero i seguenti vantaggi: • Maggiore efficienza del processo di controllo e correzione, e in definitiva del processo di produzione dei dati statistici. • Disponibilità immediata delle regole di controllo, fondamentali per il monitoraggio in tempo reale dei dati man mano rilevati e acquisiti.
Obiettivi	L'obiettivo del progetto è quello di automatizzare, il più possibile, la fase di derivazione delle regole di compatibilità, al fine di migliorare l'efficienza del processo di controllo e correzione, e in definitiva dell'intero processo di produzione statistica.
Metodologia	Le macro-fasi di una indagine statistica, esemplificando e evidenziando quanto concerne le regole di compatibilità o incompatibilità, sono di seguito modellate secondo il linguaggio ArchiMate e secondo il modello standard GSBPM v5.1 (https://unece.org/sites/default/files/2023-11/GSBPM%20v5_1.pdf).

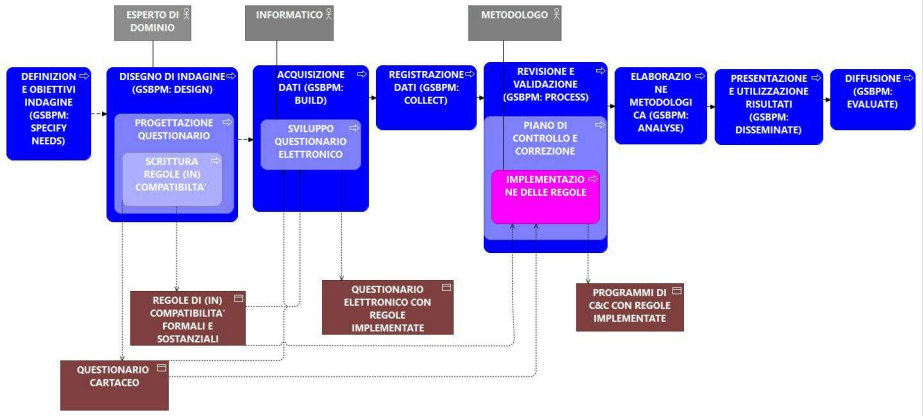

Figura 1. Macro-fasi di un'indagine statistica con evidenziazione delle fasi del processo riguardanti le regole di compatibilità.

Come evidenziato in Figura 1, le regole di compilazione sottostanti il questionario sono di input sia al processo di sviluppo del questionario elettronico sia al processo di implementazione delle regole di compatibilità nella fase di controllo e correzione.
Questa organizzazione implica la replica di una attività molto dispendiosa che potrebbe essere superata inserendo una fase di traduzione automatica delle regole.
A tal fine, si prospettano due scenari:

1) Point-to-Point
2) Rule Formal Language

Scenario 1: Point-To-Point

Nel primo scenario, chiamato **Point-To-Point**, la fase di traduzione automatica delle regole di compatibilità si interpone tra il questionario elettronico e la fase di controllo e correzione. In tal caso, è necessario implementare un traduttore per ogni sistema/linguaggio del questionario elettronico e ogni linguaggio della fase di controllo e correzione (es. Gino-R, Panda-R, Gino-SAS, …) come mostrato nello schema seguente:

La Figura 2 mostra come cambierebbe il processo di produzione statistica. In sostanza, è presente una nuova sottofase di processo, "Traduzione delle regole", nell'ambito della fase "Implementazione delle regole".

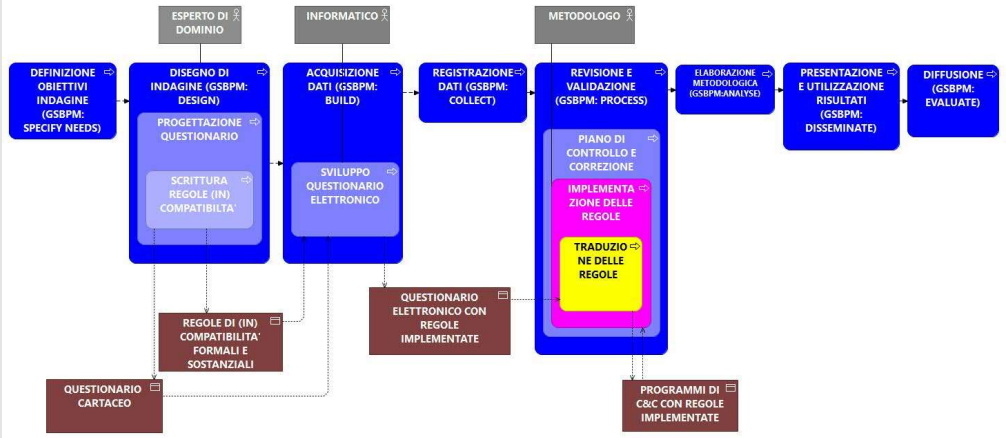

Figura 2. Processo di produzione statistica nello scenario POINT-TO-POINT.

Nell'ambito del progetto è stato implementato il traduttore Panda–R riutilizzando parte del *software* di Panda che implementa l'analisi sintattica del file XML e la sua trasformazione in oggetti Java. Partendo da tale artefatto, è stata implementata la traduzione di oggetti Java in istruzioni R.
Il traduttore traduce i seguenti tipi di quesiti del questionario:

- domanda_singola
- domanda_singola_j
- domanda_si_no
- domanda_multipla
- domanda_multipla_j
- domanda testuale
- domanda numerica
- quesiti tabellari.

Non è stata implementata la traduzione dei seguenti tipi di quesiti:

- domanda_tabella_2col_combo
- tabella_j

essendo casi di quesiti complessi e di raro utlizzo nelle indagini.

Il traduttore è stato sperimentato con il questionario del Censimento permanente delle Imprese, Anno 2022.
Il questionario presenta 183 quesiti così suddivisi per tipologia e numero:

- domanda_testuale: 10
- domanda_multipla: 8
- domanda_si_no: 14
- domanda_numerica: 5
- tabella_singola_j: 74
- domanda_singola: 4
- domain: 8
- domanda_multipla_j: 27

- tabella_multipla_j: 6
- domanda_singola_j: 21
- domanda_tabella_2col_combo: 1(non gestita)
- tabella_j: 5 (non gestite)

Il numero di tipologie tradotte e la relativa percentuale sono riportati nella tabella seguente:

Tipologia regola/quesito	**N.**	**%**
Tradotta	177	96,7
Non tradotta	6	3,3
totale	183	100

In base a quanto sopra evidenziato, le tipologie di regole residuali non tradotte possono essere facilmente integrate manualmente nel processo di controllo e correzione.
Seguono alcuni esempi di traduzione di quesiti implementati in Panda nelle corrispondenti regole R, rispettivamente input e output del traduttore.

Questi a risposta singola: domanda si_no	
Domanda del questionario	**SEZIONE 1 – PROPRIETÀ, CONTROLLO E GESTIONE** **1.1 Ad oggi l'impresa è, direttamente o indirettamente, controllata da una persona fisica o una famiglia?** **1.** Sì 2. No C1_01=0 se SI; 1 se NO
Rappresentazione XML della domanda secondo la sintassi di Panda	<elemento id="c1_01"> <type>domanda_si_no</type> <required /> <requiredMessage>c1_01_req</requiredMessage> </elemento>
Codice R corrispondente	! is.na(c1_01) & c1_01 >= 0 & c1_01 <= 1

Questi a risposta singola: Domanda Singola	
Domanda del questionario	**1.2 Chi ha la responsabilità della gestione dell'impresa?** Solo se 1.1=0 1. L'imprenditore o il socio principale o unico 2. Un membro della famiglia proprietaria o controllante 3. Un manager selezionato all'interno dell'impresa 4. Un manager assunto all'esterno dell'impresa 5. Altro soggetto
Rappresentazione XML della domanda secondo la sintassi di Panda	<elemento id="c1_02"> <hide>NOT c1_01__0</hide> <repeat>5</repeat> <type>domanda_singola</type> <required /> </elemento>
Codice R corrispondente	if (! (c1_01 == 0)) (is.na(c1_02)) if (c1_01 == 0) (! is.na(c1_02) & c1_02 >= 0 & c1_02 <= 4)

Domande a risposta aperta: Domanda numerica	
Domanda del questionario	**2.8 Tra settembre e dicembre 2022 quale è stata la percentuale del personale dell'impresa che ha lavorato a distanza, in smart working o in telelavoro, sul totale del personale?** Indicare una stima della quota % del personale dell'impresa che ha lavorato a distanza, in smart working o in telelavoro almeno un giorno a settimana. |_|_|_| %
Rappresentazione XML della domanda secondo la sintassi di Panda	<elemento id="c2_08"> <hide>!c2_07__1 AND !c2_07__2 AND !c2_07__3 AND !c2_07__4</hide> <type>domanda_numerica</type> <max>100</max> <min>0</min> </elemento>
Codice R corrispondente	if (!(c2_07 == 1) & !(c2_07 == 2) & !(c2_07 == 3) & !(c2_07 == 4)) (is.na(c2_08)) if ((!(!(c2_07 == 1) & !(c2_07 == 2) & !(c2_07 == 3) & !(c2_07 == 4))) & (! is.na(c2_08))) ((c2_08 >= 0 & c2_08 <= 100))

Scenario 2: Rule Formal Language

Il secondo scenario, chiamato **Rule Formal Language** (RFL), prevede la definizione di un linguaggio formale condiviso che gli esperti di dominio dovrebbero adottare per la scrittura delle regole durante la fase di progettazione del questionario.

Le regole scritte in tale linguaggio formale costituiscono l'input di traduttori automatici che consentono agli esperti dello sviluppo del questionario elettronico di ottenere in automatico le procedure per il sistema di acquisizione dati utilizzato (es. GINO, Panda, etc.). Analogamente, le regole scritte in linguaggio formale costituiscono l'input di traduttori automatici che producono le regole nella forma del linguaggio *software* utilizzato per la fase di controllo e correzione (es. R, SAS, etc.). A tal proposito, si veda lo schema sottostante, in cui si

comprende che le regole scritte secondo il linguaggio RFL costituiscono l'input per la traduzione automatica a seconda degli scopi.

La Figura 3 mostra come cambierebbe il processo di produzione statistica: come si può osservare emergono due sottofasi di processo, "Traduzione RFL–Linguaggio sistema" e "Traduzione RFL–Linguaggio C&C", rispettivamente in "Sviluppo del questionario elettronico" e "Implementazione delle regole".

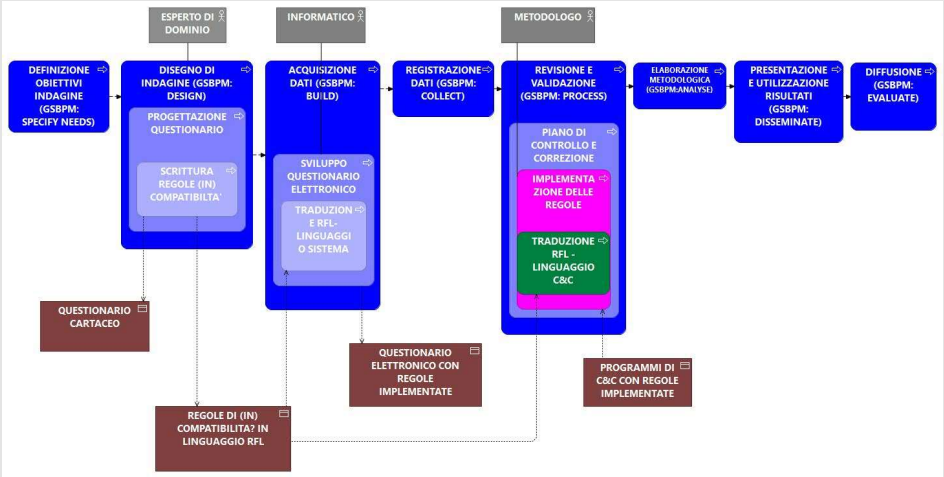

Figura 3. Processo di produzione statistica nello scenario RULE FORMAL LANGUAGE.

Una possibile realizzazione di questo scenario prevede che le regole scritte in RFL siano depositate in un archivio comune, ad esempio MetaStat, come parte integrante della documentazione del processo di indagine a cui riferirsi. Tuttavia, l'archiviazione in MetaStat non è strettamente necessaria, è sufficiente un foglio di lavoro o tabella opportunamente organizzati.
Ai fini della definizione del RFL è stata effettuata un'analisi preliminare del linguaggio VTL (*Validation and Transformation Language*), sviluppato nell'ambito della comunità SDMX per la standardizzazione della manipolazione dei dati (validazioni e trasformazioni), e dei pacchetti VTL Engine ed Editor sviluppati da Banca d'Italia. Da tale analisi è emersa un'elevata complessità del linguaggio VTL, eccessiva rispetto a quanto necessario per definire le regole di compatibilità, che andrebbe a complicare notevolmente il procedimento di traduzione automatica delle regole.
Escluso l'utilizzo di VTL, le scelte successive hanno portato alla definizione di un linguaggio formale semplice ed intuitivo che consente di esprimere tutte le possibili tipologie di regole.

Il linguaggio che si propone ha le seguenti caratteristiche:

1. Ciascuna regola finisce con il carattere ";"
2. I nomi di variabile possono contenere le lettere minuscole a-z, le lettere maiuscole A-Z, i numeri 0-9, i caratteri speciali _, -
3. Sono parole chiave del linguaggio:
   - **NOT** esprime la *negazione* di una condizione;
   - **AND** esprime *and logico* tra due condizioni semplici;
   - **OR** esprime *or logico* tra due condizioni semplici;
   - **NULL** rappresenta il valore *missing*;
   - **!=** rappresenta l'operatore *not-equal*;
   - **==** rappresenta l'operatore *confronto di uguaglianza (equal)*;
   - **=** rappresenta l'assegnazione;
   - **<, <=, >, >=** operatori di *minore, minore-uguale, maggiore, maggiore-uguale* applicabili a variabili quantitative;
   - **IN** operatore applicabile a variabili qualitative (ad es. regole di dominio);
   - **IF-THEN** esprimono l'istruzione condizionale;
   - **LENGTH** rappresenta l'operatore di lunghezza di una stringa;
   - **+,-,/,*** rappresentano le operazioni numeriche tra variabili numeriche;
   - **JUMP** operatore che consente di esprimere il salto ad un quesito (es. IF (var1==NULL) {JUMP quesito 10;}.

4. I costrutti base del linguaggio sono:
   - **Condizione semplice**: è formata da una variabile, un operatore di confronto e una variabile o una costante. Esempi:
     - c1_01 >= 0; usata per esprimere parte del campo di validità di una variabile quantitativa;
     - c1_1<=c1_2;
     - C2 IN (0,1,2); usata per esprimere il dominio di validità di una variabile qualitativa (nell'esempio, la variabile c1_02 ha valori ammissibili (A=0, B=1, C=2)).
   - **Istruzione IF-THEN:** esprime una regola condizionale. La sintassi è la seguente:

     IF (condizione)
     THEN {lista istruzioni se condizione vera;}

     Esempio 1 di regola condizionale:

     ```
     IF (c1_01 = 0 OR c1_01 = NULL)
       THEN {c1_02 >=0;
             c1_02 <=4;}
     ```

     Esempio 2 di regola condizionale:

     ```
     IF (c1_01 = 0 OR c1_01 = NULL)
       THEN {c1_02 >=0;
     ```

c1_02 <=4;
c2=5;}

NON sono ammessi blocchi IF-THEN annidati.
Inoltre, non è previsto il ramo ELSE in quanto la regola IF-THEN-ELSE può essere sempre riformulata con due regole IF-THEN nel seguente modo:

IF (cond1)
THEN {istruzione1;}
ELSE {istruzione2;}
si riscrive nelle seguenti due regole:
IF(cond1)
THEN {istruzione1;}
IF(NOT cond1)
THEN {istruzione2;}

Nella seguente tabella vengono riportati alcuni esempi di scrittura nel linguaggio RFL di alcune regole del questionario dell'Indagine Multiscopo sulle Aziende Agricole, Anno 2025.

Quesito-variabile	Descrizione	Regola
D3_1	Negli ultimi cinque anni, l'azienda ha realizzato investimenti finalizzati a innovare la tecnica e/o la gestione della produzione?	IF (D0_2==1 OR D03==1) THEN {D3_1 IN (1:2);};
D3_2	Indicare se gli investimenti innovativi sono stati fatti nei seguenti ambiti:	IF (D3_1 == 2) THEN {JUMP (D3_2a, D3_2b, D3_2c, D3_2d, D3_2e, D3_2f, D3_2g); };
D3_4	L'azienda ha deciso di effettuare investimenti innovativi in…	If (D3_1==1) THEN {NOT (D3_4a==2 AND D3_4b==2 AND D3_4c==2 AND D3_4d==2);};
D3_15	Tra 12 mesi quale sarà approssimativamente la percentuale della SAU con strumenti dell'agricoltura 4.0?	IF (D3_14==1) THEN {D3_15 > D3_13;};

Conclusioni

Lo scenario **Point-To-Point** presenta i seguenti aspetti positivi e negativi:

- Positivi:
  - Realizzabile senza ulteriori vincoli.

	• Negativi:
	  ○ Un diverso traduttore per ogni sistema/*software* del questionario elettronico e per ogni linguaggio *software* della fase di controllo e correzione.
	  ○ Potenziali errori sono ereditati dalle fasi successive dell'intero processo.
	  ○ Complessità del linguaggio/*software* in cui è sviluppato il questionario elettronico.
	
	Lo scenario **Rule Formal Language** presuppone i seguenti elementi chiave:
	• Visione di insieme
	• Condivisione
	• Collaborazione
	• Standard di processo
	
	I benefici attesi sono molteplici; tra questi si elencano:
	• Efficienza del processo.
	• Riduzione costi in termini di tempo e risorse umane.
	• Automatizzazione.
	• Agevole manutenzione.
	• Riduzione degli errori indotti.
	• Ridotta complessità.
	• Replicabilità del metodo.

Risultati ottenuti	Il progetto si propone di rendere più efficace e soprattutto più efficiente la fase di definizione delle regole di compatibilità all'interno di un processo di controllo e correzione. A tal fine vengono proposti due nuovi scenari di organizzazione del processo di produzione statistica: (1) scenario Point-To-Point; (2) scenario Rule Formal Language. Con riferimento al primo scenario (PtP) è stato realizzato un traduttore puntuale da Panda a R, dove Panda rappresenta il questionario elettronico, mentre R è il *software* in cui si sviluppano le procedure di controllo e correzione. Nel secondo scenario, invece, è stata individuata una formulazione del linguaggio formale (RFL) che gli esperti del fenomeno indagato dovrebbero adottare per esplicitare le regole che definiscono il questionario di rilevazione. A partire da esse, sia lo sviluppo del questionario elettronico sia la definizione delle regole di compatibilità sono processi che possono essere automatizzati, rendendo così più efficace ed efficiente l'intero processo di produzione statistica.
Membri del Team	Adele M. Bianco, Simona Rosati, Laura Tosco (DIRM/DCME/MEA) Luigi Arlotta (DIRM/DCIT/ITE) Renato Torelli (DIRM/DCRD/RDO)