

UNIVERSITÀ DEGLI STUDI DI MODENA
E REGGIO EMILIA

Facoltà di Ingegneria - Sede di Modena
Corso di Laurea in Ingegneria Informatica

Progetto e realizzazione
di un catalogo virtuale per e-commerce
con il sistema MOMIS

Relatore
Chiar.mo Prof. Sonia Bergamaschi

Tesi di Laurea di
Stefano Neri

Correlatore
Ing. Alberto Corni

Controrelatore
Chiar.mo Prof. Enrico Scarso

Anno Accademico 1999 - 2000

Parole chiave:
catalogo virtuale
e-commerce
intelligent information integration
mediatore
internet

RINGRAZIAMENTI

Un sentito ringraziamento va alla Professoressa Sonia Bergamaschi per l'aiuto che mi ha fornito durante la realizzazione della presente tesi e per la costante disponibilità dimostrata.

Vorrei inoltre ringraziare tutti i componenti del team MOMIS, in particolare l'Ing. Alberto Corni, per i consigli ed i chiarimenti di ordine pratico ed implementativo.

Un grazie di cuore ai miei genitori che col loro affetto e le loro rinuncie mi hanno permesso di arrivare a questo traguardo.

Un ultimo ringraziamento a Laura e a tutte le persone che mi sono state vicine in questi anni.

Indice

1	Il commercio elettronico	1
1.1	Cos'è l'e-commerce	1
1.2	E-commerce vs commercio tradizionale	1
1.2.1	Business to consumer (B2C)	3
1.2.2	Business to business (B2B)	8
1.3	I vantaggi dell'e-commerce	10
1.4	Gli svantaggi dell'e-commerce	11
1.5	I limiti allo sviluppo dell'e-commerce	12
2	I linguaggi dell'e-commerce e l'evoluzione del Catalogo Virtuale	15
2.1	Il Catalogo Virtuale	15
2.2	L'HTML	16
2.2.1	La Vetrina	16
2.3	Common Gateway Interface (CGI)	20
2.3.1	Vantaggi e svantaggi delle CGI	21
2.4	Java	22
2.4.1	Le caratteristiche di Java	23
2.5	Il Catalogo Virtuale oggi	25
2.6	L'eXtensible Markup Language (XML)	26
2.6.1	L'XML e lo scambio dei dati	27
2.7	I sistemi di integrazione intelligente delle informazioni	29
2.7.1	Il Mediatore	29
2.7.2	I limiti dei sistemi I^3	31
2.7.3	Il Multicatalog	32
3	L'integrazione intelligente con MOMIS	35
3.1	Il sistema MOMIS	35
3.1.1	Il linguaggio ODL_{I^3}	36
3.1.2	Gli strumenti di MOMIS	37
3.2	Il processo d'integrazione	39
3.3	Generazione del Thesaurus comune	40

3.3.1	Estrazione delle relazioni intra-schema	40
3.3.2	Estrazione delle relazioni inter-schema	41
3.3.3	Arricchimento dell'insieme delle relazioni	41
3.3.4	Validazione delle relazioni	41
3.3.5	Inferenza di nuove relazioni	42
3.4	Il calcolo delle affinità	42
3.5	La generazione dei cluster	43
3.6	La costruzione delle classi globali	43
3.7	Fusione degli attributi	44
3.7.1	Fusione degli attributi di relazioni validate	44
3.7.2	Fusione di attributi di relazioni non validate	45
3.8	La generazione delle mapping table	45
4	SI-Designer: il tool per la creazione dello schema globale	47
4.1	SI-Designer in MOMIS	47
4.2	L'automa di utilizzo di SI-Designer	47
4.3	Source	50
4.4	SIM A	51
4.5	SLIM	51
4.6	SIM B	53
4.7	EXTM Rel	54
4.8	Cluster	55
4.9	TUNIM	57
5	Il catalogo virtuale realizzato con il sistema MOMIS	61
5.1	Obiettivi	61
5.2	Le fonti	62
5.2.1	Volkswagen	62
5.2.2	Renault	67
5.2.3	Fiat	70
5.3	I wrapper	74
5.4	SI Designer	87
5.4.1	L'uso di WordNet	89
5.4.2	L'arricchimento del progettista	117
5.4.3	I cluster	119
5.4.4	Le mapping table	122
5.5	Conclusioni	188
A	Il linguaggio ODL³	189

Elenco delle figure

1.1	Diagramma dei tipi di commercio	2
1.2	Passaggi del processo di acquisto	4
1.3	Passaggi del processo di vendita	5
1.4	Passaggi del processo del business to business	9
2.1	Pagina web di presentazione di un listino di computer	17
2.2	Procedura di funzionamento delle CGI	21
2.3	Schema della Java Base Platform	23
2.4	Confronto dello sviluppo del codice con sistemi CGI e Java	24
2.5	Scambio di dati per mezzo di schema XML	28
2.6	Architettura dei Sistemi I^3	30
2.7	Architettura multicatalog	32
3.1	Architettura del sistema MOMIS	36
3.2	Struttura di ODB Tools	38
3.3	Fasi del processo di integrazione con MOMIS	40
3.4	Esempio di mapping table	45
4.1	Architettra del sistema MOMIS	48
4.2	Automa di funzionamento di SI-Designer	49
4.3	Pannello Source	50
4.4	Pannello Thes.Rel: run SIM	52
4.5	Pannello SLIM	53
4.6	Pannello Hypernym	54
4.7	Pannello EXTM Rel	55
4.8	Pannello Cluster: ARTEMIS Configuration	56
4.9	Pannello Cluster: Cluster Tuning	57
4.10	Pannello TUNIM	58
4.11	Pannello TUNIM: Schema Visualizer	59
5.1	Diagramma E\R del database Volkswagen	63
5.2	Diagramma E\R del databse Renault	67

5.3	SI Designer: i wrapper	88
5.4	SI Designer: ARTEMIS	120

Capitolo 1

Il commercio elettronico

1.1 Cos'è l'e-commerce

L'e-commerce, commercio elettronico, così come definito dal Ministero dell'Industria, del Commercio e dell'Artigianato, *consiste nello svolgimento di attività commerciali e di transazioni per via elettronica e comprende attività diverse quali: la commercializzazione di beni e servizi per via elettronica, la distribuzione on-line di contenuti digitali, l'effettuazione per via elettronica di operazioni finanziarie e di borsa, ...*

Le strutture che hanno permesso la nascita di questo tipo di commercio e che ne sostengono lo sviluppo sono le reti di computer, prima tra tutte Internet. È quindi facile comprendere come le sorti del commercio elettronico siano fortemente legate allo sviluppo e alla diffusione dei computer e di Internet; e di come le sue peculiarità e i suoi difetti siano dipendenti dalle problematiche di una rete globale e per molti versi senza regole.

Lo studio dell'e-commerce va certamente svolto in relazione e in confronto al commercio tradizionale, ma va soprattutto sviluppato tenendo conto di tutte le tecnologie informatiche ed elettroniche che lo possono rendere un proficuo business, ma che ne sono in certi casi il freno e la limitazione alla crescita.

1.2 E-commerce vs commercio tradizionale

Il commercio tradizionale, così come tutti lo conoscono, può essere definito come uno scambio di prodotti, di servizi e di denaro tra due parti, e comprende tutte le azioni necessarie a completare la transazione.

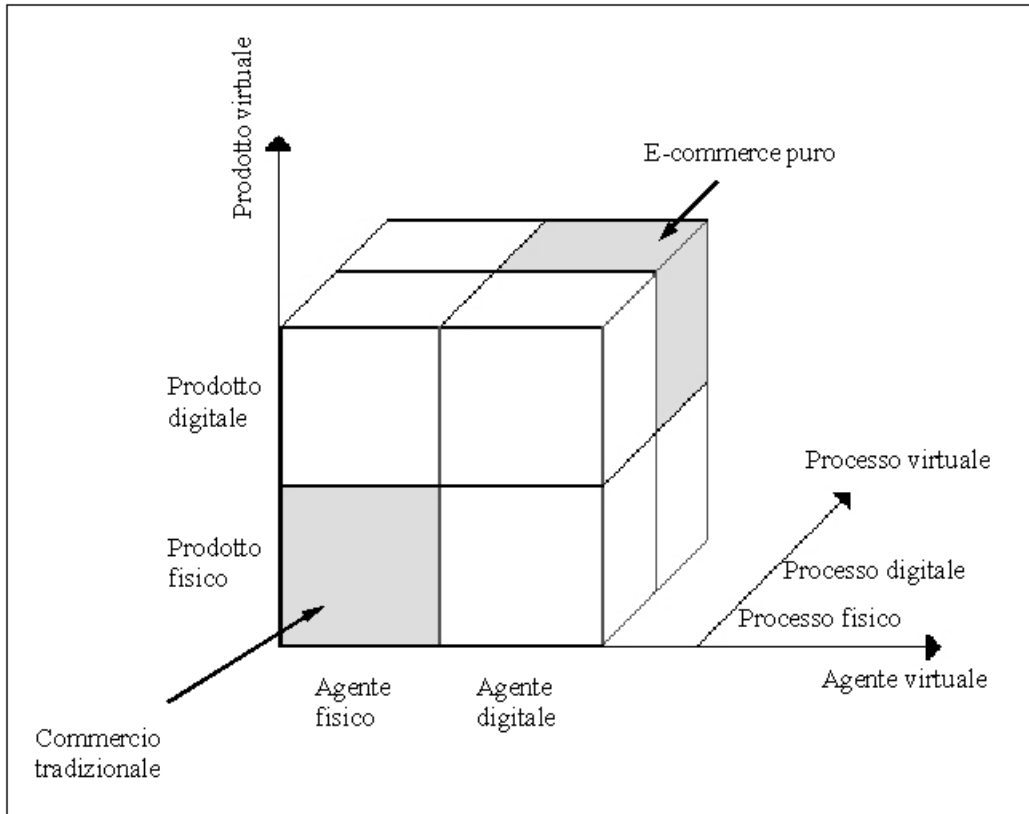


Figura 1.1: Diagramma dei tipi di commercio

È facile rilevare dalle definizioni date di commercio tradizionale e di e-commerce, come queste due attività abbiano punti in comune, ma anche sostanziali differenze; è quindi possibile creare una classificazione in funzione delle diverse fisicità e digitalità degli elementi che competono e caratterizzano il commercio in generale: prodotto o servizio, processo, intermediario.

Come si può osservare in figura 1.1 [1], il commercio tradizionale più strettamente definito è quello in cui tutti gli elementi di valutazione sono fisici, si ha: un prodotto materiale, un passaggio di mano in mano del prodotto e del denaro, e un commesso che attende al processo di vendita.

L'e-commerce puro è invece quello caratterizzato da componenti digitali: il prodotto è un software o comunque costituito da un insieme di file, il processo è automatico, cioè gestito da un programma, all'interno del quale è poi possibile trovare un assistente virtuale che guida e aiuta nelle varie fasi della transazione.

Queste caratteristiche fisiche e digitali si possono combinare per creare i diversi tipi di e-commerce misto.

Per meglio cogliere le differenze e le potenzialità che l'e-commerce ha nei confronti del commercio tradizionale è necessaria un'analisi dettagliata delle diverse tipologie:

- business to consumer,
- business to business.

1.2.1 Business to consumer (B2C)

Il B2C è quella forma di commercio in cui beni e servizi vengono venduti all'utente finale, il consumatore; è quell'attività a cui ognuno di noi prende parte, come cliente, svariate volte in una giornata, acquistando il pane, il giornale, un vestito o un'automobile.

Confrontiamo pertanto le dinamiche che il B2C ha nel commercio tradizionale e in quello elettronico, sia dal punto di vista del cliente che da quello del venditore.

Prendiamo in considerazione, come esempio, la vendita/acquisto di un'auto-mobilità. La dinamica di acquisto [2], come evidenziato in figura 1.2, attraversa i seguenti punti:

- l'esigenza che una persona ha di un'automobile;
- la ricerca di un'auto, con determinate caratteristiche, presso i concessionari di zona;
- il confronto dei modelli, delle offerte e delle diverse condizioni di vendita e di garanzia;
- la contrattazione col rivenditore sulle modalità di consegna e di pagamento;
- ritiro dell'auto e pagamento;
- attivazione della garanzia.

Passiamo quindi dalla parte del concessionario e vediamo quali attività compie nel processo di vendita, sempre per quel che riguarda il commercio tradizionale:

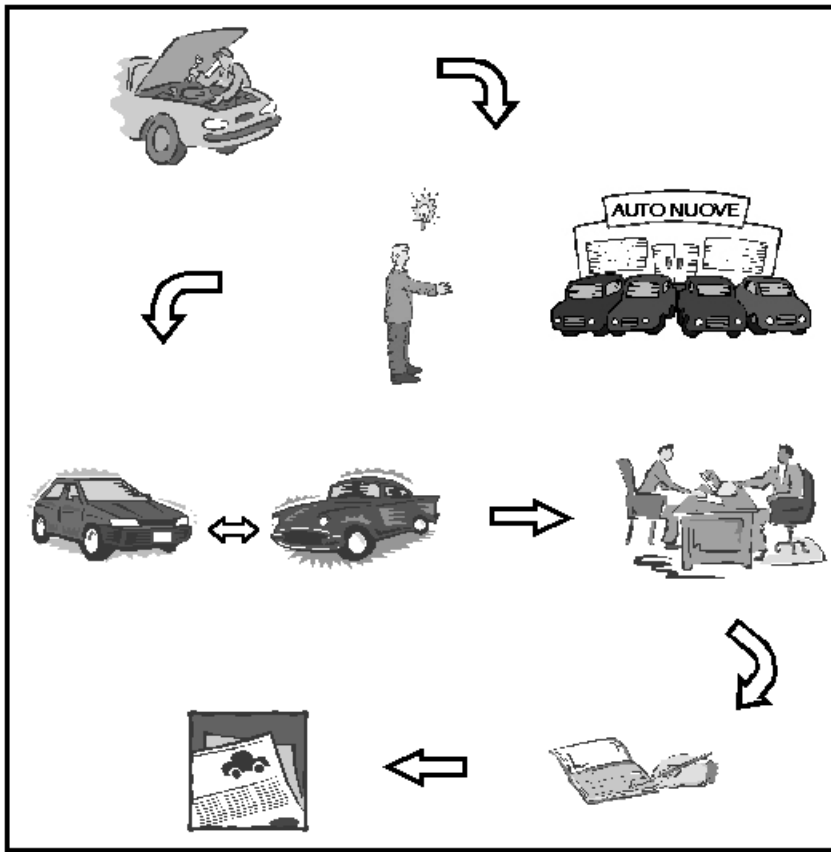


Figura 1.2: Passaggi del processo di acquisto

- compie un'indagine di mercato per individuare le esigenze dei possibili clienti;
- si rifornisce di quelle auto che più incontrano i gusti e le necessità del pubblico;
- crea offerte per contrastare i concorrenti del settore;
- realizza una campagna di marketing con cartelloni pubblicitari e inserzioni sui giornali locali per far conoscere le proprie offerte e occasioni;
- mostra e fa provare le proprie auto ai clienti;
- contratta con l'acquirente le modalità di consegna e di pagamento;
- consegna l'auto e riscuote il denaro;

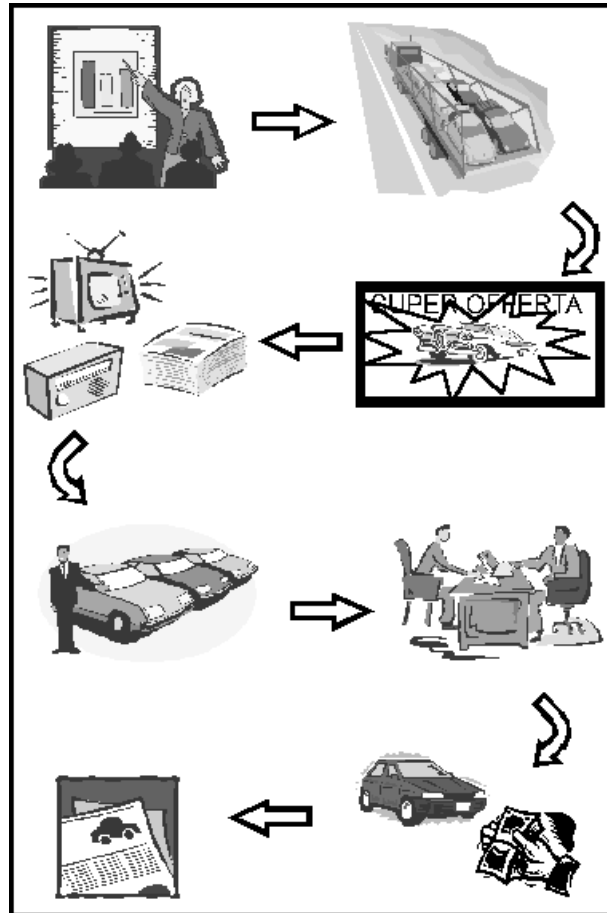


Figura 1.3: Passaggi del processo di vendita

- attiva la garanzia sul prodotto venduto.

Le dinamiche dell'e-commerce non hanno una sequenza di punti particolarmente diversa da quelle appena viste, ma sono certamente diversi i modi in cui vengono attuati.

Analizziamo le azioni del cliente:

- ha l'esigenza di un'automobile, con precisi requisiti - questo è lo stesso punto di partenza per un acquisto nel commercio tradizionale;

- accende il computer, si collega a Internet e naviga tra i siti delle case produttrici d'auto e dei concessionari - la differenza a questo punto diventa notevole: il cliente compie la ricerca, del bene di cui ha bisogno, senza muoversi da casa, godendo della possibilità di accedere direttamente anche alle offerte delle case produttrici, aumentando così il panorama delle offerte tra cui scegliere. A questo punto, però, si manifesta anche un problema: se non si conoscono gli URL precisi dei rivenditori e dei produttori, ma ci si affida ai motori di ricerca si rischia una grossa perdita di tempo nella sequela di link spesso insignificanti, se non del tutto vuoti;
- confronta le schede tecniche e le offerte che ha trovato;
- una volta fatta la scelta si ricollega al sito con l'offerta preferita, e seguendo la procedura realizzata dal rivenditore, seleziona il modello con tutti i particolari che lo interessano, quindi le modalità di consegna e di pagamento - tutto ciò avviene ancora una volta nella massima comodità per il cliente che fa ogni cosa stando davanti al proprio computer.
È da sottolineare, inoltre, come questo processo venga svolto in maniera oggettiva e automatica, nella totale assenza di interazioni umane, con gli aspetti positivi e negativi che questo comporta.
A questo livello del processo si deve poi sottolineare il secondo significativo ostacolo: la fiducia del cliente sulla sicurezza del web nella protezione di informazione e dati personali relativi al pagamento;
- ritira l'auto, che gli può essere consegnata direttamente a casa, e paga secondo la modalità prescelta sul web;
- attiva la garanzia.

Questi punti evidenziano come superando alcuni piccoli problemi e un po' di incertezza il cliente possa guadagnare tempo, qualità e diversificazione di prodotti, dall'uso dell'e-commerce.

Mentre le cose si fanno relativamente più comode e semplici per l'acquirente, la realizzazione del commercio elettronico complica inizialmente la dinamica del rivenditore.

Per passare da un impianto per il commercio tradizionale a uno che realizzi quello elettronico il venditore deve prima di tutto costruire un sistema web che si integri con il sistema informativo già esistente nell'azienda, questo per avere una struttura che opera in modalità real-time interfacciandosi col magazzino, col sistema gestionale e con quello di spedizione.

Questo sistema deve, inoltre, garantire sicurezza e la possibilità di gestire le diverse forme di pagamento:

- off line (contrassegno, bonifico, bollettini postali, ...);
- on line (carte di credito, attivazione di pagamenti e mutui presso banche presenti in Internet).

Il sito costruito deve essere facile da navigare, con un'interfaccia accattivante, con un catalogo ampio, ma soprattutto con prodotti descritti esaurientemente e presentati, anche graficamente, in tutti i loro aspetti, per poter superare il limite della virtualità dell'articolo dettata dal web.

Una volta realizzata la parte tecnologica il commerciante ha la necessità di farsi conoscere, e lo può fare attraverso forme di comunicazione tradizionali, quali giornali, radio e televisione, oppure con metodi più innovativi: cd card, cd rom pubblicitari, link dai portali, banner nei siti più importanti e più visitati, oppure iscrivendosi ai motori di ricerca che, però, con la crescita esponenziale dei siti web forniscono ogni giorno di più elenchi che non riescono a soddisfare con precisione la richiesta dell'utente.

Quando con i passaggi precedenti si sono costruite le basi per l'implementazione dell'e-commerce, riprende il processo che abbiamo visto per il commercio tradizionale, con strumenti che lo velocizzano e lo potenziano:

- il commerciante raccoglie dati sulle esigenze e le preferenze dei clienti - questo può avvenire in maniera esplicita, per esempio attraverso questionari, oppure registrando e analizzando dati che il cliente comunica in modo implicito come il suo paese di provenienza, la lingua scelta, il software che utilizza, i link che visita e quindi a che tipo di prodotti e a che modelli è più interessato;
- utilizzando lo strumento di marketing appena descritto, per raccogliere informazioni, l'azienda può realizzare una strategia di vendita tesa a soddisfare le esigenze più diffuse: acquista i modelli delle auto più ricercate e realizza su queste delle offerte che attraggano i clienti e che possano contrastare la concorrenza - quest'ultimo è un aspetto molto significativo, infatti se da un lato l'e-commerce permette a un'azienda di raggiungere un maggior numero di clienti grazie a una rete di comunicazione globale, dall'altro la pone in concorrenza con più aziende di quante non ce ne fossero nell'ambito locale del commercio tradizionale;

- il rivenditore realizza, quindi, una campagna pubblicitaria, sfruttando gli stessi strumenti usati per farsi conoscere e in più inviando messaggi personalizzati ai clienti che si sono registrati nel sito;
- le altre fasi del processo: vendita, consegna dell'auto, riscossione del denaro, attivazione della garanzia vengono relizzati in automatico dal sistema informativo dell'azienda e dal software che gestisce il sito web per il commercio elettronico.

Riassumendo si può dire che dal confronto con la dinamica del commercio tradizionale emerge come l'e-commerce richieda un maggiore lavoro a monte del processo di vendita per implementare e per sfruttare le tecnologie proprie di questa attività, ma semplifichi e velocizzi poi tutto il resto del procedimento; concretizzando, una volta ammortizzata la fase di avvio, un notevole vantaggio in termini di tempo, di diffusione e quindi di guadagni rispetto al precedente.

1.2.2 Business to business (B2B)

Come B2B [2] viene classificato il commercio tra soggetti appartenenti a una stessa catena produttiva, anche detta catena del valore: produttori, fornitori, negozianti, ...

In figura 1.4 si possono seguire, come esempio, i passaggi relativi alla realizzazione di un motore per automobili:

- vengono estratte le materie prime, i minerali;
- i materiali vengono poi raffinati per eliminare le scorie;
- i metalli purificati sono poi fusi per realizzare le varie parti del motore: pistoni, testata, biella, ...;
- si assemblano i componenti componenti, completando la costruzione del motore;
- il motore terminato giunge in ultimo alla casa produttrice di auto che lo monta sui suoi modelli.

Questo processo prima dell'avvento dei computer e soprattutto delle reti che li collegano era, ovviamente, tutto manuale, con tutti gli inconvenienti che questo comportava.

Gli ordini di approvvigionamento dovevano sottostare alla lentezza, nonchè ai

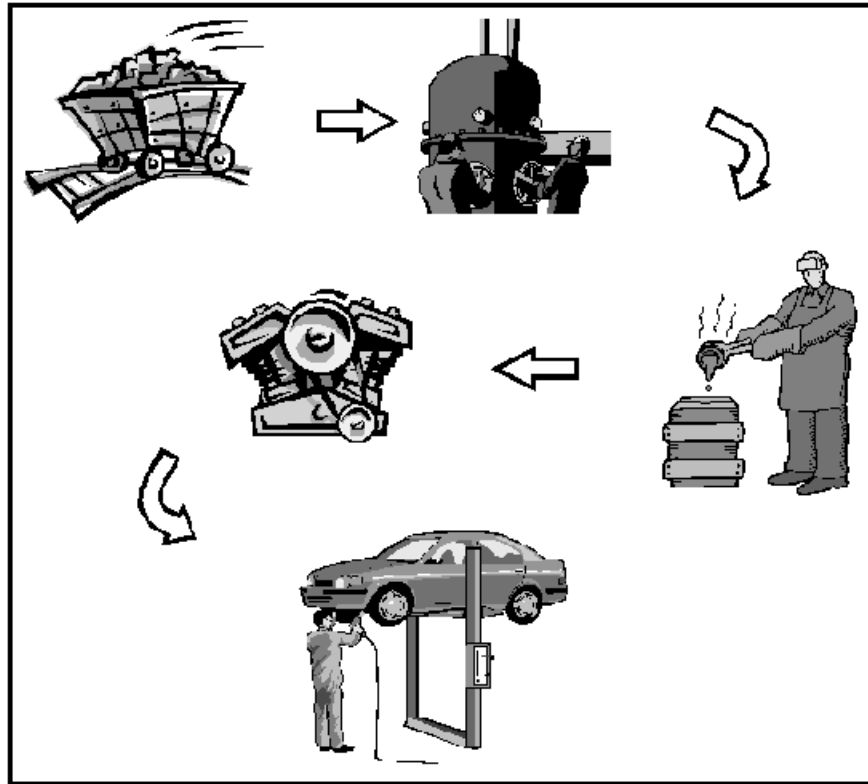


Figura 1.4: Passaggi del processo del business to business

costi, dei sistemi di consegna a mano (corrieri, posta, ...) e ai ritardi provocati dalle burocrazie industriali non coordinate da sistemi automatici.

Conseguenza diretta del punto precedente era la staticità del processo dovuta appunto alla presenza di considerevoli tempi morti tra la richiesta dei beni e la soddisfazione della stessa.

Per supplire alla latenza tra richieste e consegne si realizzavano approvvigionamenti consistenti per sostenere le esigenze di medio e lungo termine, dell'azienda. Quest'ultima strategia, sebbene cercasse di risolvere un problema, ne ha poi creati dei nuovi ugualmente gravi. La prima questione da risolvere era legata alle elevate spese di mantenimento delle scorte in magazzino; il secondo problema era invece la poca malleabilità nei confronti dei mutamenti nella domanda del mercato: se la domanda aumentava più del previsto si creavano ritardi nelle consegne con conseguenti aumenti dei prezzi che potevano anche portare alla perdita dei clienti; se la domanda calava o si esauriva poichè i consumatori richiedevano prodotti diversi, si concretizzava l'esigenza di smaltire le giacenze di prodotti e materie già lavorate presenti in magazzino; eliminazione che veniva realizzata nel migliore

dei casi con vendite a prezzi ribassati, che producevano guadagni bassi se non nulli, oppure nel peggiore dei casi con la distruzione delle stesse con consistente perdita per l'impresa.

Il primo passo fatto per porre rimedio a questi problemi è stata l'introduzione di sistemi informativi che velocizzassero le attività all'interno delle aziende.

Successivamente a questa innovazione, e grazie all'introduzione di mezzi di comunicazione tra computer, nacque l'esigenza di connettere i sistemi informativi delle imprese appartenenti alla stessa catena produttiva, al fine di rendere più veloce e dinamico l'intero processo.

Quello appena descritto non è stato un passaggio banale e ha creato una problematica che è attuale più che mai a causa della realizzazione del mercato globale: l'integrazione di sistemi informativi diversi e personalizzati che ha come conseguenza lo scambio di dati non sempre ugualmente strutturati.

Questo problema poteva essere risolto relativamente a ogni singola catena produttiva, a cui aderivano un limitato numero di aziende, con l'introduzione di interfacce software dedicate, anche se costose; soluzione che non è però attuabile se si vuole allargare la catena del valore per sfruttare le migliori opportunità che si hanno di volta in volta a disposizione grazie al mercato globale realizzato tramite Internet.

1.3 I vantaggi dell'e-commerce

Come visto nei paragrafi precedenti la realizzazione di una attività di e-commerce ha notevoli vantaggi, poichè supera i limiti spaziali e temporali tipici del commercio tradizionale.

Innanzitutto il commercio elettronico avviene in un regime di mercato globale, attraverso Internet, non ci sono quindi più ostacoli legati allo spazio: un commerciante può rivolgere le sue offerte non solo alle persone che vivono nelle vicinanze del suo negozio, ma anche a tutti coloro che tramite la rete possono accedere al suo negozio virtuale. Analogamente avviene per i clienti che oltre i classici negozi possono accedere alle proposte di commercianti di tutto il mondo.

Come avviene nel B2C così anche nel B2B l'e-commerce permette di modificare e allargare le catene produttive con l'introduzione di nuove aziende che offrono garanzie e prodotti migliori, e con le quali si instaurano rapporti tramite il web. Il secondo beneficio che porta il commercio elettronico è relativo al tempo:

- i negozi virtuali sono aperti ventiquattrore su ventiquattro, vantaggio per i commercianti, ma soprattutto per i clienti che sono liberi di fare acquisti quando più lo desiderano e gli fa comodo;

- grazie alla connessione a Internet i dati vengono trasmessi istantaneamente, questo non provoca attese nel B2C, ma più importante tolgono ritardi nel B2B: le aziende con l'uso di un avanzato sistema gestionale possono modificare giornalmente le strategie produttive variando le quantità di approvvigionamenti e quelle dei prodotti, riducendo così i rischi conseguenze dei ritardi dei tradizionali mezzi di comunicazione.

Un altro vantaggio che si presenta alle imprese che realizzano questo nuovo tipo di commercio è la possibilità di offrire i propri prodotti direttamente ai clienti finali senza passare per intermediari e negozianti al dettaglio.

Tutti questi elementi che generano benefici in termini di velocità e di comodità, permettendo di svolgere tutte le pratiche stando seduti davanti al proprio computer, concorrono a determinare un enorme vantaggio in ambito economico:

- i commercianti aumentano i loro guadagni ampliando il numero dei clienti;
- le aziende che vendono i loro prodotti ai clienti finali possono distribuirli a prezzi superiori a quelli fatti ai negozianti, pur offrendo la merce a costi più economici di quelli del mercato tradizionale;
- il cliente può trovare una maggiore varietà di prodotti a prezzi più vantaggiosi di quelli riscontrabili nei soliti negozi.

1.4 Gli svantaggi dell'e-commerce

Sebbene, come abbiamo visto, l'e-commerce, una volta attuato, produca notevoli benefici, soprattutto in termini economici, si devono però sottolineare anche gli inconvenienti che lo caratterizzano.

Il primo problema che si evidenzia è dettato dalla virtualità del sistema che determina l'impossibilità di dare un volto al rivenditore o al cliente; e che impedisce al cliente di toccare e verificare materialmente il bene che vuole acquistare.

Un secondo svantaggio, dovuto alla realizzazione di una attività sovranazionale, risiede nel dover superare le barriere culturali e quelle legali. Soprattutto quest'ultime sono rilevanti per l'uso di una rete di comunicazione dalle regole non sempre chiare e della quale non si riesce facilmente a identificare un garante.

L'ultimo problema riscontrabile è la difficoltà, per le imprese che vogliono aprire una attività di e-commerce, di calcolare se e in quanto tempo gli investimenti possono essere recuperati e che guadagni ne seguiranno.

1.5 I limiti allo sviluppo dell'e-commerce

L'e-commerce è in una fase di grande diffusione, ma trova ancora degli ostacoli alla sua espansione.

Un primo scoglio, che si incontra nel B2C, è rappresentato dal computer, che per la difficoltà di essere usato non riesce a essere accettato come strumento fondamentale in tutte le case. Il superamento di questo problema è affidato ad apparecchi più semplici e maggiormente diffusi, quali i telefoni cellulari.

Il passaggio da una piattaforma potente e complessa, quale un pc, a una piccola e relativamente semplice è anch'esso un limite poichè comporta un cambiamento nel modo di trasferire e formattare i dati, per avere comunicazioni veloci e risultati presentabili sui ridotti schermi dei telefoni portatili.

Un'altra barriera alla diffusione del commercio elettronico la si trova nella limitata larghezza di banda per la trasmissione dei dati.

Nel B2B si è sempre avuta la possibilità di usufruire di tecnologie più potenti di quelle di comune accesso: linee dedicate ad alta velocità, collegamenti satellitari, il nuovo sistema ADSL; il tutto chiaramente a prezzi proporzionati alle prestazioni e non certo accessibili da un semplice utente Internet.

Nel B2C la larghezza di banda è ancora uno spinoso problema in quanto le pagine commerciali sono sempre più complesse, per essere attraenti, e questo implica la trasmissione di una consistente quantità di dati. Questi lunghi tempi necessari alla visualizzazione delle pagine web hanno due effetti negativi: innanzitutto il cliente si può stancare di aspettare e quindi abbandonare le ricerche e ritornare al commercio tradizionale; in secondo luogo il tempo passato in collegamento a Internet comporta una spesa telefonica che può portare l'utente ad avere ripensamenti su come spende il tempo e soprattutto il proprio denaro.

L'ostacolo più significativo è certamente legato alla sicurezza sempre precaria, a causa di una rete di comunicazione di libero accesso, anche ai malintenzionati.

Il problema della sicurezza costringe le imprese a costosi investimenti, senza ritorno, per l'acquisto e il costante aggiornamento di sistemi di protezione.

Le violazioni alla sicurezza delle aziende, perpetrate dagli hackers, hanno come conseguenza la perdita di fiducia nel sistema da parte dell'utente finale, il cliente del B2C, che abbandona l'idea di fare acquisti in rete per non diffondere i propri dati e quelli della propria carta di credito su Internet. È per quest'ultimo motivo che, come abbiamo visto nell'analisi del B2C, sono ancora presenti diversi metodi di pagamento off-line.

Due inconvenienti che creano incertezza alle aziende per il passaggio all'e-commerce sono i considerevoli investimenti necessari per server aggiuntivi, rispetto a quelli già posseduti, e le spese per il software dedicato al commercio elettronico che è tanto più costoso quanto maggiormente si integra con i sistemi informativi già

presenti nell'impresa.

Capitolo 2

I linguaggi dell'e-commerce e l'evoluzione del Catalogo Virtuale

2.1 Il Catalogo Virtuale

L'elemento cardine attorno a cui è possibile costruire una applicazione di commercio elettronico è il catalogo virtuale. Come qualunque processo di acquisto anche quello realizzato via web pone il suo inizio nella selezione del prodotto che avviene confrontando i dati e le offerte che le imprese presentano nelle loro pagine web, da qui l'obbligo per ogni azienda di porre un catalogo su Internet che colpisca e soddisfi le esigenze di ogni potenziale cliente che visita il suo sito. Perché un catalogo virtuale incontri le necessità degli acquirenti deve avere particolari caratteristiche:

- semplicità: il cliente non si deve smarrire mentre analizza il catalogo, e non deve trovare difficoltà nella comprensione delle strutture e dei dati;
- dinamicità: cioè deve essere aggiornato in tempo reale sui cambiamenti delle condizioni di vendita, dei prezzi e soprattutto della disponibilità della merce;
- completezza: quindi deve presentare esaurientemente i dettagli di tutti i prodotti che l'azienda vende;
- confrontabilità con altri cataloghi: se i cataloghi delle ditte che vendono prodotti simili avessero le stesse caratteristiche, e presentassero la merce con gli stessi dati il cliente sarebbe facilitato nell'individuare quale bene soddisfa maggiormente la sue esigenze.

È evidente come questa caratteristica trovi la sua massima espressione in un catalogo unico in cui vengono raccolti e confermati i dati prelevati dai cataloghi delle singole aziende che producono gli stessi articoli. Nei prossimi paragrafi vedremo quali di queste caratteristiche siano state soddisfatte nel tempo, grazie all'evoluzione dei linguaggi utilizzati per realizzare i siti web, e come sia possibile concretizzare quelle ancora incompiute.

2.2 L'HTML

L'html [3,4], Hypertext Markup Language, nasce tra il 1989 e il 1991 per mano di due ricercatori del CERN, Tim Berners-Lee e Robert Caillau, che avevano l'esigenza di un sistema informativo interconnesso raggiungibile da un qualunque computer in uso dell'Istituto di ricerca. Berners-Lee prese come spunto di partenza l'SGML [Standard Generalized Markup Language] standard ISO dal 1986, che era un linguaggio molto complesso e vasto e probabilmente per questo poco utilizzato, lo semplificò selezionando quegli elementi che soddisfacevano le sue esigenze e diede origine alla prima versione dell'HTML. Il linguaggio venne poi standardizzato e migliorato a partire dalla versione 2.0 del 1994. L'opera di Berners-Lee ha avuto una grande diffusione grazie alla sua caratteristica più evidente: la semplicità; con pochi tag è infatti possibile formattare del testo, inserire immagini e realizzare così pagine web. Un altro elemento fondamentale che ha dato potere all'HTML come linguaggio base per Internet è l'Hyperlink cioè la possibilità di collegare le pagine web sparse sui diversi computer connessi alla rete creando così un unico grande computer globale. Grazie a questi presupposti ha avuto inizio la fortuna di Internet, che superando le barriere delle Università e degli organi statali ha raggiunto i PC di tutte le case creando appunto le fondamenta per lo sviluppo del commercio elettronico.

2.2.1 La Vetrina

Come abbiamo detto la diffusione di Internet e la semplicità dell'Html hanno avvicinato tutti gli utenti di computer all'uso della rete globale, situazione che è stata vista da molti commercianti come una nuova opportunità per il loro business. Il primissimo uso di Internet fatto dai commercianti è stato quello pubblicitario: si realizzavano pagine web allo stesso modo di come si inserivano trafiletti sulle pagine gialle, cioè con lo scopo di farsi conoscere. L'esponentiale aumento degli utenti di Internet ha spinto i rivenditori a fare un passo avanti e a dare vita alla prima vera fase dell'e-commerce: la vetrina. Nella vetrina le Aziende realizzavano un rudimentale catalogo virtuale in cui elencavano i loro prodotti e le condizioni di vendita. Vediamo per esempio come questo poteva essere realizzato da una


```

<P><CENTER><B><FONT SIZE="+2"></FONT></B>
<TABLE WIDTH="875" BORDER="1" CELLSPACING="2" CELLPADDING="0"
HEIGHT="196">
  <TR>
    <TD WIDTH="20%" HEIGHT="50">
      <P><CENTER>&nbsp;<B>Codice articolo</B></CENTER></TD>
    <TD WIDTH="20%" HEIGHT="50">
      <P><CENTER>&nbsp;<B>Nome articolo</B></CENTER></TD>
    <TD WIDTH="20%" HEIGHT="50">
      <P><CENTER><B>Caratteristiche articolo</B></CENTER></TD>
    <TD WIDTH="20%" HEIGHT="50">
      <P><CENTER><B>Prezzo unitario</B></CENTER></TD>
    <TD WIDTH="20%" HEIGHT="50">
      <P><CENTER><B>Quantit&agrave; disponibile</B></CENTER>
      </TD>
  </TR>
  <TR>
    <TD WIDTH="20%" HEIGHT="26">
      <P ALIGN=RIGHT>&nbsp;<B>1</B></TD>
    <TD WIDTH="20%" HEIGHT="26">
      PCesempio1</TD>
    <TD WIDTH="20%" HEIGHT="26">
      Primum III 700 Mhr</TD>
    <TD WIDTH="20%" HEIGHT="26">
      <P ALIGN=RIGHT>L. 2.500.000</TD>
    <TD WIDTH="20%" HEIGHT="26">
      <P ALIGN=RIGHT>&nbsp;<B>5</B></TD>
  </TR>
  <TR>
    <TD WIDTH="20%" HEIGHT="26">
      <P ALIGN=RIGHT>&nbsp;<B>2</B></TD>
    <TD WIDTH="20%" HEIGHT="26">
      PCesempio2</TD>
    <TD WIDTH="20%" HEIGHT="26">
      Primum III 500 Mhr</TD>
    <TD WIDTH="20%" HEIGHT="26">
      <P ALIGN=RIGHT>&nbsp;<B>L. 2.000.000</B></TD>
    <TD WIDTH="20%" HEIGHT="26">
      <P ALIGN=RIGHT>&nbsp;<B>4</B></TD>
  </TR>
  <TR>

```

```

    <TD WIDTH="20%" HEIGHT="26">
    <P ALIGN=RIGHT>&nbsp;3</TD>
    <TD WIDTH="20%" HEIGHT="26">
    PCesempio3</TD>
    <TD WIDTH="20%" HEIGHT="26">
    APP H7 650 Mhr</TD>
    <TD WIDTH="20%" HEIGHT="26">
    <P ALIGN=RIGHT>&nbsp;L. 2.400.000</TD>
    <TD WIDTH="20%" HEIGHT="26">
    <P ALIGN=RIGHT>&nbsp;6</TD>
  </TR>
  <TR>
    <TD WIDTH="20%" HEIGHT="25">
    <P ALIGN=RIGHT>&nbsp;4</TD>
    <TD WIDTH="20%" HEIGHT="25">
    PCesempio4</TD>
    <TD WIDTH="20%" HEIGHT="25">
    APP H7 500 Mhr</TD>
    <TD WIDTH="20%" HEIGHT="25">
    <P ALIGN=RIGHT>&nbsp;L. 2.100.000</TD>
    <TD WIDTH="20%" HEIGHT="25">
    <P ALIGN=RIGHT>&nbsp;3</TD>
  </TR>
  <TR>
    <TD WIDTH="20%" HEIGHT="26">
    <P ALIGN=RIGHT>&nbsp;5</TD>
    <TD WIDTH="20%" HEIGHT="26">
    PCesempio5</TD>
    <TD WIDTH="20%" HEIGHT="26">
    Velos 500 Mhr</TD>
    <TD WIDTH="20%" HEIGHT="26">
    <P ALIGN=RIGHT>&nbsp;L. 1.700.000</TD>
    <TD WIDTH="20%" HEIGHT="26">
    <P ALIGN=RIGHT>&nbsp;5</TD>
  </TR>
</TABLE></CENTER></P>
<P>&nbsp;</P>
<P><CENTER><B>E-mail: soc_vend_pc@azienda.com</B></CENTER>
</BODY></HTML>

```

Il codice evidenzia come inserire anche solo una tabella di cinque righe fosse un

lavoro oneroso; questo però non era l'unico problema ce n'era uno ancora più importante: la staticità delle pagine Html. Mentre i dati contenuti nel database del sistema informativo dell'azienda venivano aggiornati in funzione della vendita e degli acquisti, quelli presentati sulle pagine web dell'impresa, essendo distaccati e aggiornati solo saltuariamente, venivano spesso a trovarsi in uno stato di incoerenza rispetto alle reali disponibilità del rivenditore. Vediamo quali problemi tutto ciò comportava alla realizzazione dell'e-commerce: un cliente visita un sito e trova la disponibilità di determinati prodotti a precise condizioni di vendita, decide quindi per l'acquisto e manda un e-mail o un fax all'azienda: per effetto dell'incoerenza dei dati del sito si può quindi veder rispondere che il prodotto non è più disponibile, o che è variato il prezzo, A questo punto, il cliente può ripetere l'iter per un prodotto differente, cercare su Internet altre ditte che vendono il prodotto che lo interessa oppure stanco di perdere tempo rivolgersi al mercato tradizionale. A queste condizioni l'e-commerce non avrebbe mai avuto una grande possibilità di espandersi, era necessario dinamicizzare il processo con un sistema che collegasse i dati del sito con quelli aggiornati in tempo reale sul sistema informativo dell'azienda per presentare ai clienti dati veritieri sulla disponibilità di prodotti dell'azienda; il primo tentativo in questo senso è stato realizzato tramite le CGI.

2.3 Common Gateway Interface (CGI)

Le CGI sono "porte" che permettono ai Web server di accedere ed eseguire altri programmi e di incorporare il loro output, testo, grafica e audio, nelle pagine inviate ai web browser. È evidente come l'utilizzo di programmi, svincolati dalla staticità dell'html, con l'utilizzo di variabili, parametri, ... offra un notevole dinamismo a tutto il sistema; dinamismo che si manifesta nelle possibilità di realizzare oltre alle animazioni dei siti, applicazioni in real-time e quindi sistemi di commercio elettronico consistenti.

I linguaggi utilizzati per realizzare CGI si possono classificare in due tipi:

- compilati, come il C e il C++;
- interpretati, come il PERL.

Le differenze fra queste due classi sono sostanziali e significative. La prima classe offre come vantaggio dei programmi compilati una volta per tutte prima del loro utilizzo che consentono così una risposta veloce alle chiamate del server; ma è caratterizzato anche da alcuni aspetti negativi; i linguaggi compilati sono infatti molto complessi da imparare e da utilizzare, ed è difficile la manutenzione e la modifica dei loro programmi.

I linguaggi interpretati hanno caratteristiche antitetiche a quelle dei linguaggi

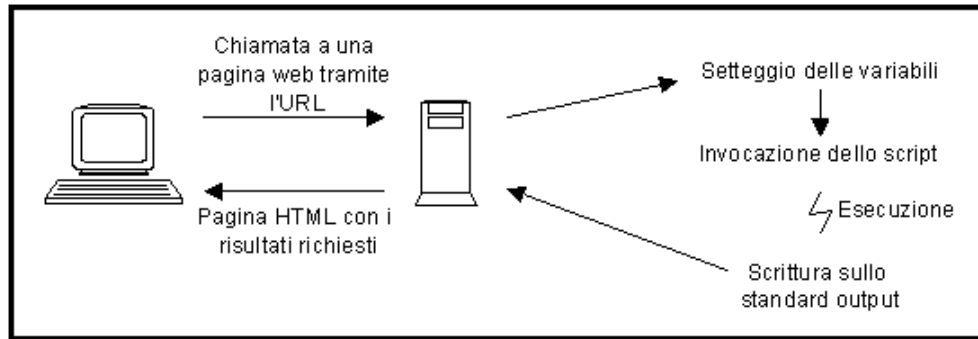


Figura 2.2: Procedura di funzionamento delle CGI

compilati: sono script che devono essere interpretati tutte le volte che vengono invocati, il che implica tempi di risposte più lenti e un maggiore impegno per il server che deve ricompilare svariate volte lo stesso script; inoltre sono relativamente più semplici da imparare, ma certamente più facili da modificare. Vediamo ora come funzionano le CGI.

Il browser decodifica la prima parte dell'URL e contatta il server al quale fornirà poi la seconda parte dell'URL in cui è contenuta la chiamata a una CGI. Il server traduce l'URL nel nome di un file, riconosce che non è un file statico, ma uno script CGI, quindi setta una determinata serie di variabili di stato che passa, insieme con i dati e le richieste ricevute dal browser, ad esempio una select SQL per un determinato database, al programma che viene lanciato. Lo script va in esecuzione, legge le variabili di stato e il file di standard input in cui sono contenute le altre informazioni, compie le azioni richieste - si interfaccia al database, esegue la select e riceve i dati - quindi scrive i risultati sullo standard output, invia la corretta intestazione MIME al server e termina. Il server rilevata la conclusione dello script manda i risultati formattati al browser che li visualizzerà, e chiude la connessione. Il processo è evidentemente più complesso e oneroso, in termini delle risorse hardware necessarie, rispetto al banale recupero di informazioni HTML.

2.3.1 Vantaggi e svantaggi delle CGI

Il vantaggio, certamente non trascurabile, prodotto dalle CGI è quello di avere introdotto nel web nuovi elementi dinamici che permettono l'esecuzione di transazioni in real-time e la possibilità di interfacciarsi non solo con e-elementi HTML, ma anche con dati in continua evoluzione come quelli dei database aziendali. Questa innovazione portata dalle CGI ha dato la spinta per la realizzazione di un vero e proprio commercio elettronico in cui i processi di vendita avvenissero in maniera

consistente e in modo immediato.

Bisogna però anche sottolineare gli svantaggi delle CGI, che fanno sostanzialmente capo alla loro caratteristica di risiedere e girare sul server:

- impegnano spazio sulla memoria fissa del server;
- quando sono in funzione sfruttano le risorse hardware del server;
- pongono problemi alla sicurezza.

I limiti più consistenti sono certamente gli ultimi due.

Se vengono invocati contemporaneamente più script o uno stesso script un svariato numero di volte, si possono creare problemi di saturazione nell'uso della CPU e della memoria che si traducono in consistenti ritardi nelle risposte, quando proprio non provocano un blocco totale del sistema.

Il secondo grosso limite delle CGI, soprattutto di quelle compilate, è dato dalla sicurezza: le CGI possono avere accesso a file residenti sulla memoria permanente del server e possono in certi casi modificarli o cancellarli per finalità dolose.

Questi problemi hanno dato la spinta per la ricerca di un sistema che funzionasse maggiormente dal lato client e ponesse meno problemi di sicurezza; è sotto queste ipotesi che è esploso il fenomeno JAVA.

2.4 Java

Java è un linguaggio di programmazione sviluppato alla Sun Microsystem nel 1991 da parte di James Goslin all'interno del progetto Green.

Il progetto Green aveva l'obiettivo di sviluppare software per i dispositivi elettronici tipici di ogni casa (TV, tostapane, impianti di illuminazione, ...), cioè per sistemi embedded, in modo che questi potessero comunicare tra loro.

Nel progetto Green venne quindi realizzato un prototipo, chiamato Star 7, una specie di dispositivo di controllo che in modo remoto potesse dialogare con altri sistemi dello stesso tipo. Il tentativo di sviluppare Star 7 in C++ provocò dei blocchi al lavoro e per questo motivo Goslin si decise a creare un linguaggio che meglio si adattasse alle sue esigenze, doveva essere:

- di piccole dimensioni;
- efficiente;
- portabile sui più svariati dispositivi;
- affidabile.

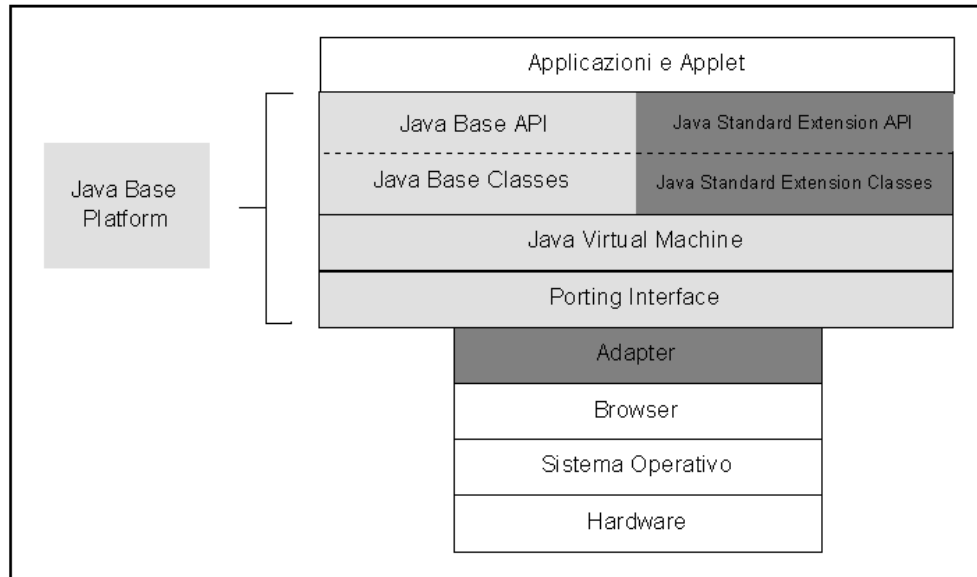


Figura 2.3: Schema della Java Base Platform

Queste caratteristiche di Java si sono rivelate le chiavi fondamentali per il suo utilizzo nel web e fu così che nel 1994 venne creato il primo web browser, chiamato Hotjava, in grado di eseguire programmi java, le così dette applet.

Il vero decollo di java come linguaggio per il web si ebbe però solo dal 1995 quando Netscape ne acquistò la licenza e lo inserì nel suo più diffuso browser.

2.4.1 Le caratteristiche di Java

La prima caratteristica di Java è la sua portabilità, permessa dal fatto che i suoi programmi e le sue applet vengono eseguiti in un ambiente dedicato, detto Java platform, che per quanto riguarda le applet deve essere presente nel browser.

La Java platform, la cui struttura si può osservare in figura 2.3, ha come elementi fondamentali la Java Virtual Machine (JVM) e la Java Application Programming Interface (Java API).

La JVM è un computer virtuale realizzato via software, capace di adattarsi ai diversi microprocessori esistenti; si interfaccia col browser grazie al Porting Interface che ha una parte indipendente dalla piattaforma e una dipendente chiamata Adapter.

Il Porting Interface e gli Adapter permettono il trasporto della JVM su diversi sistemi senza l'obbligo di riscriverlo completamente ognivolta.

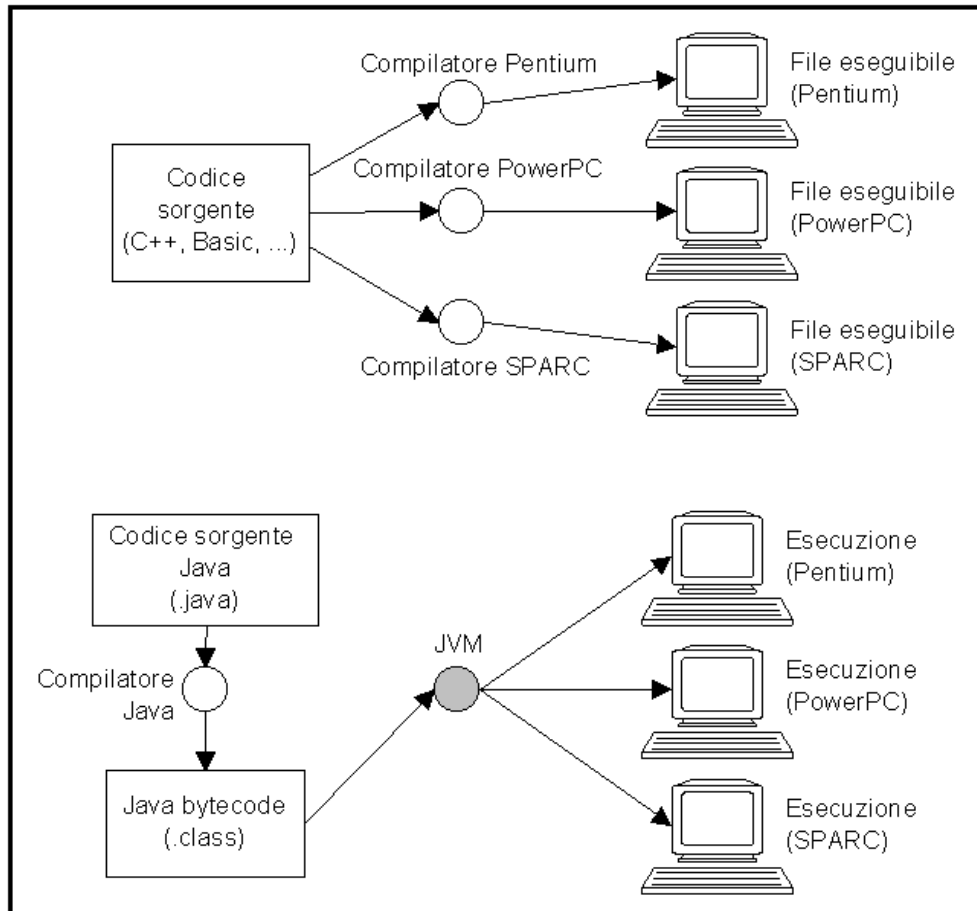


Figura 2.4: Confronto dello sviluppo del codice con sistemi CGI e Java

La Java API rappresenta invece un'interfaccia standard verso le applicazioni e le applet, indipendente dall'architettura su cui è fatta girare.

Grazie all'esistenza di questa struttura si è potuto far sì che il compilatore Java non producesse un codice binario specifico e direttamente eseguibile su una particolare macchina, ma producesse un codice intermedio, chiamato bytecode, indipendente dalla piattaforma, che sarebbe poi stato interpretato dalla JVM.

In figura 2.4, si può osservare la differenza tra un qualunque altro linguaggio di programmazione ad alto livello e Java, nel passaggio del codice da sorgente a eseguibile.

Le caratteristiche di Java che abbiamo visto sinora suppliscono ampiamente ad alcune delle carenze evidenziate dalle CGI, pur conservandone le proprietà di dinamicità:

- è un linguaggio più semplice rispetto al C++;
- è indipendente dalla piattaforma, viene scritto e compilato una sola volta per tutti i sistemi;
- gira dal lato client, alleggerendo così il lavoro del server.

Inoltre Java pone rimedio anche al più grosso problema creato dalle CGI: la mancanza di sicurezza.

Java, infatti, è stato progettato in modo tale da garantire la massima sicurezza contro l'esecuzione di programmi dannosi: ogni programma scaricato dalla rete viene racchiuso in un "sandbox" e può operare solo al suo interno, non può quindi leggere o scrivere sul disco locale, connettersi a host remoti, creare altri processi, caricare dinamicamente librerie o metodi nativi che avrebbero accesso alle risorse del sistema.

2.5 Il Catalogo Virtuale oggi

Grazie alle CGI, ai linguaggi di script e soprattutto alle innovazioni e alla sicurezza introdotta da Java il catalogo virtuale si è fortemente dinamicizzato passando dallo stato di vetrina a quello di "scaffale" da cui i clienti possono prelevare i prodotti in un commercio elettronico reale; il tutto senza eccessive complicazioni nella sua realizzazione.

Vediamo infatti, per esempio, come in Java servano poche righe di codice per accedere ai database e operare su di essi le transazioni e le richieste necessarie alle operazioni di e-commerce:

```
1)      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
  
2)      Connection con = DriverManager.  
          getConnection("jdbc:odbc:nome_db");  
  
3)      Statement stmt = con.createStatement();  
  
4.1)    ResultSet rs = stmt.executeQuery("Select ...");  
  
4.2)    stmt.executeUpdate("Insert (Delete) ...");
```

Innanzitutto si carica il driver java (1), ci si connette al database (2), quindi si crea uno statement (3) e si operano le operazioni sql (4.1 select, 4.2 insert, delete,

update). I risultati ottenuti vengono poi formattati per essere presentati all'interno dell'applet.

Con questi strumenti e con i potenziamenti operati dalle grandi aziende produttrici di software nei loro pacchetti per il commercio elettronico tutte le imprese hanno avuto modo di aprire un loro negozio virtuale e quindi un'attività di e-commerce. Giunti a questo stato dell'arte, però non tutti i problemi dell'commercio elettronico sono stati risolti, ci sono ancora due importanti questioni in sospeso:

1. l'impossibilità di realizzare multicatalog a causa delle diverse strutture utilizzate nei singoli cataloghi, che si traduce nella difficoltà per un cliente di trovare e di confrontare i dati riguardanti un prodotto che lo interessa per acquistare quello che viene venduto al prezzo più conveniente;
2. la difficoltà per un'azienda di fare interfacciare il proprio sistema informativo con quello delle diverse società da cui si rifornisce al fine di scambiare dati.

Gli approcci utilizzati per la soluzione di questi problemi sono stati essenzialmente due:

- l'utilizzo dell'XML;
- l'utilizzo di un mediatore per l'integrazione di sorgenti di dati eterogenee.

2.6 L'eXtensible Markup Language (XML)

L'XML [5] nasce nel 1996 per mano dell'XML Working Group (originariamente noto come SGML Editorial Review Board) che col benestare del W3C aveva l'obiettivo di definire un nuovo metalinguaggio di tipo descrittivo e di facile utilizzo, che permettesse la definizione di un set di tag personalizzati per la rappresentazione dei dati, la salvaguardia delle strutture dati per opera del Document Type Definition (DTD), il file in cui vengono descritte, e la possibilità di validare i documenti operando un confronto con le regole descritte nel DTD.

Queste proprietà che caratterizzano l'XML hanno spianato la strada alla possibilità di raccogliere e presentare i dati in modo ottimizzato su diversi supporti (browser internet, cellulari, ...) e sistemi informativi.

Il primo elemento caratteristico dell'XML è il markup ossia il tag che precede e segue il dato caratterizzandolo:

```
<tag> dato </tag>
```

esempio:

```
<nome> Stefano </nome>
```

Il significato del tag si trova nel DTD, che descrivendo la struttura del documento definisce ogni tag:

```
<!ELEMENT [nome_elemento]([valore])>
```

esempio:

```
<!ELEMENT nome (#PCDATA)>
```

I valori permessi per gli elementi descritti sono:

- #PCDATA: per le stringhe di caratteri;
- ANY: per qualunque tipo;
- una lista di altri elementi;
- EMPTY: nel caso l'elemento sia vuoto.

Con questi strumenti è possibile realizzare un catalogo virtuale in XML; catalogo che può apparire per molti versi simile a uno realizzato in HTML, ma ha il vantaggio che chiunque accedendo al DTD può ricostruire una struttura organizzata dei dati e quindi convertirli con un tool dedicato nel formato che più gli è comodo, ad esempio in un database relazionale.

2.6.1 L'XML e lo scambio dei dati

Le caratteristiche dell'XML che abbiamo evidenziato non offrono una risposta immediata al primo problema lasciato in sospeso dall'attuale catalogo virtuale, cioè la possibilità di realizzare multicatalog, ma strutturando i dati semplifica il problema di integrazione di sorgenti dati diverse.

Al contrario le proprietà dell'XML hanno posto le condizioni necessarie per tentare di superare gli ostacoli nella scambio dei dati tra i sistemi informativi delle aziende.

L'obiettivo più completo che può essere proposto è quello di uno standard per gli schemi delle strutture dati e per i tag che li identificano.

Una volta realizzato e applicato uno standard in cui i dati sono identificati univocamente per mezzo del loro tag, in strutture ben definite, diventa banale la realizzazione di software che catturi e manipoli i dati.

È evidente, però, che il progetto è molto complesso visto che gli elementi della

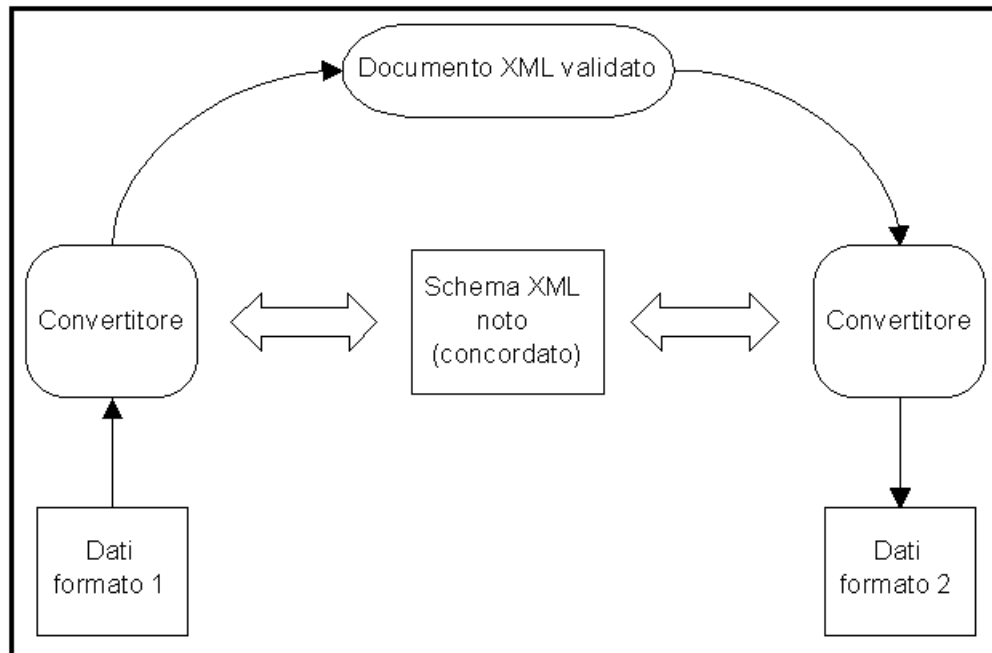


Figura 2.5: Scambio di dati per mezzo di schema XML

realtà che dovrebbero essere rappresentati tramite i loro tag sono praticamente infiniti.

È già più semplice la realizzazione di schemi standardizzati, e magari concordati, per la rappresentazione delle strutture dati maggiormente diffuse e utilizzate.

Un passo in questo senso è stato fatto col progetto Biztalk framework [6] lanciato da Microsoft in associazione con svariate altre imprese.

Il progetto, indirizzato soprattutto alle aziende, ha lo scopo di raccogliere gli schemi delle strutture XML dei dati delle società che vogliono partecipare e di renderle consultabili alle imprese interessate, in questo modo le aziende che collaborano accedendo agli schemi raccolti, e magari concordandone di nuovi, possono più facilmente scambiarsi le informazioni.

Dalla situazione descritta si evidenzia però che l'uso dell'XML rimane una strada di compromesso perchè per funzionare obbliga tutte le aziende al suo uso e alla conversione dei propri dati, conservati nelle strutture più comode per i rispettivi sistemi informativi (database relazionali o a oggetti, documenti di testo, ...), in elementi di un documento XML.

L'alternativa a questa scelta è l'uso di un sistema di integrazione intelligente delle informazioni, il cui onere ricade solamente sulle aziende interessate alla raccolta dei dati.

2.7 I sistemi di integrazione intelligente delle informazioni

I sistemi di integrazione delle informazioni I^2 [7] sono strumenti che sfruttando la descrizione dei dati di diverse sorgenti, o parti di esse, riescono a raccogliarli e combinarli tra loro.

Al fine di una significativa e corretta fusione delle informazioni si esige che l'integrazione venga fatta con "intelligenza", per questo motivo i sistemi iniziali si sono evoluti in sistemi di integrazione intelligente delle informazioni I^3 , che fanno uso di tecniche di Intelligenza Artificiale, capaci di arricchire il valore dei dati prima di combinarli.

L'Intelligent Integration of Information [8] è un progetto attivo dal 1992, sostenuto dall'ARPA (Advanced Research Project Agency, una agenzia del Dipartimento della Difesa statunitense), che sulla base di uno standard mira a porre le basi per i servizi dei sistemi di integrazione intelligente di sorgenti di dati eterogenee; in modo da ridurre costi di realizzazione e manutenzione di supersistemi che gestiscano grandi quantità di dati di raccolte non correlate.

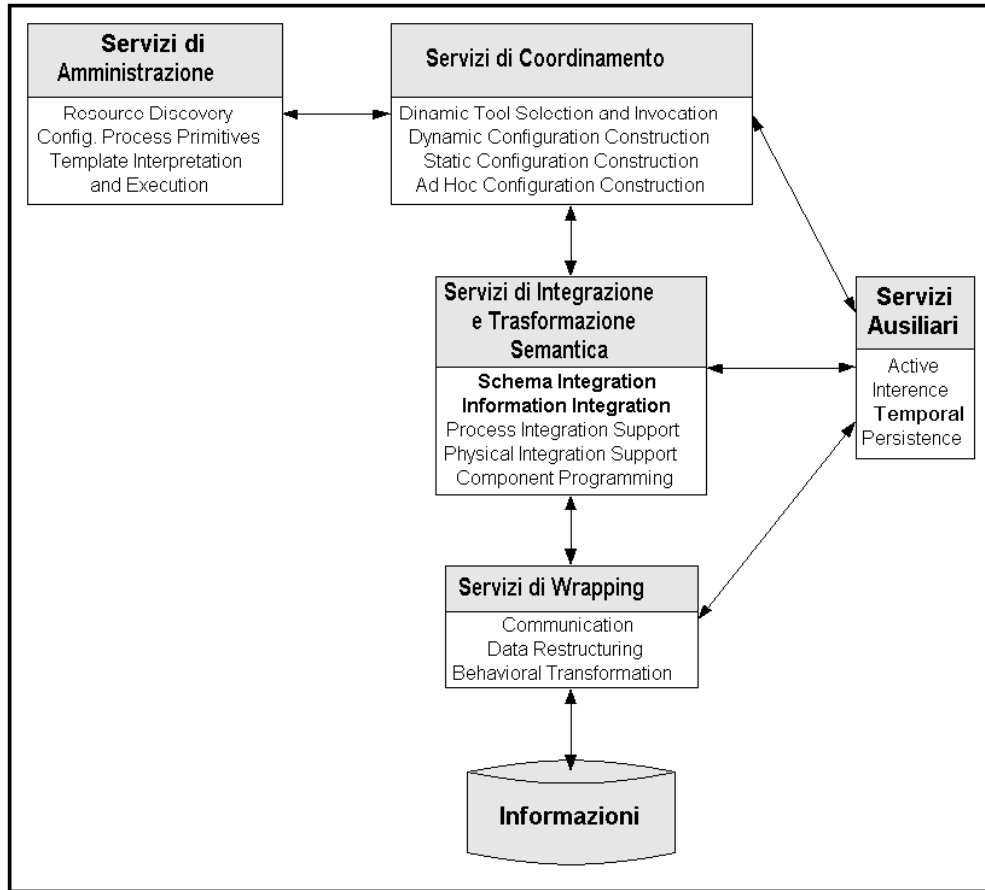
L' I^3 si inserisce nei servizi dei supersistemi in posizione centrale tra gli utenti e i dati, con diversi moduli:

- **Mediatore e Facilitatore:** raccolgono e combinano i dati dalle fonti selezionate;
- **Query Processor:** rielabora le query per ottimizzarle;
- **Data Miner:** estrae informazioni intensionali implicite dall'analisi dei dati.

2.7.1 Il Mediatore

Il Mediatore [9] è un modulo del sistema di integrazione intelligente delle informazioni e si inserisce a livello intermedio dell'architettura di riferimento I^3 , ed esattamente all'interno dei Servizi di Integrazione e Trasformazione Semantica, ai quali sono assegnati i compiti di:

- integrazione degli schemi;
- integrazione delle informazioni derivanti da più sottoquery, per fornire un'unica risposta alla richiesta iniziale;
- supporto al processo di integrazione, in particolare nella suddivisione di una query in molteplici sottoquery al fine di ottimizzare la ricerca.

Figura 2.6: Architettura dei Sistemi I³

Come si osserva in figura 2.6, i Servizi di Integrazione e Trasformazione Semantica si trovano tra i Servizi di Coordinamento e Amministrazione, che selezionano le sorgenti e i servizi che incontrano le esigenze dei clienti, e i Servizi di Wrapping, l'interfaccia tra i sistemi di integrazione e le singole sorgenti, che traducono i dati eterogenei in un linguaggio standard.

In parallelo a quelli già citati si trovano poi i Servizi Ausiliari, che arricchiscono il sistema con diverse funzioni per il monitoraggio, la propagazione degli aggiornamenti, l'ottimizzazione e l'arricchimento semantico dei dati.

I compiti del mediatore sono più precisamente:

- gestire le eterogeneità delle diverse fonti di informazioni;
- integrare i dati;
- riportare i dati all'utente che li richieda;

- garantire la stabilità del servizio.

2.7.2 I limiti dei sistemi I^3

Le potenzialità dei sistemi I^3 sono enormi: la raccolta e la gestione automatica o semiautomatica di dati da sorgenti eterogenee.

Questi strumenti, però, non hanno ancora raggiunto la completezza a causa di due importanti limiti [10,11,12]:

- i problemi ontologici;
- i problemi semantici.

I problemi ontologici nascono dal fatto che nella descrizione dei dati non tutte le fonti condividono i concetti fondamentali dei tre livelli di ontologia:

1. *top-level ontology*: in cui vengono descritti gli elementi più generali (spazio, tempo, ...);
2. *domain and task ontology*: in cui si descrive il vocabolario relativo a un dominio (medicina, giurisprudenza, ...) o a una attività (vendita, costruzione, ...);
3. *application ontology*: nella quale sono descritti i concetti dipendenti sia dal dominio che dall'obiettivo.

Al momento attuale i sistemi I^3 operano delle semplificazioni per aggirare alcuni limiti ontologici, il sistema Momis, che verrà analizzato nel capitolo 3, ad esempio, ipotizza che le fonti da integrare appartengano a uno stessa Domain Ontology.

I problemi di semantica insorgono per il fatto che pur considerando fonti che descrivono una medesima realtà, queste sono state modellate da persone diverse e quindi presentano:

- eterogeneità tra classi di oggetti, per l'uso di nomi, attributi o metodi diversi;
- eterogeneità tra le strutture delle classi, per l'uso di diversi criteri di specializzazione o di aggregazione, o per discrepanze semantiche;
- eterogeneità delle istanze delle classi, per le diverse unità di misura usate per i domini di uno stesso attributo.

Questo tipo di problemi comportano un oneroso lavoro, non automatico, di analisi dei dati per una corretta integrazione.

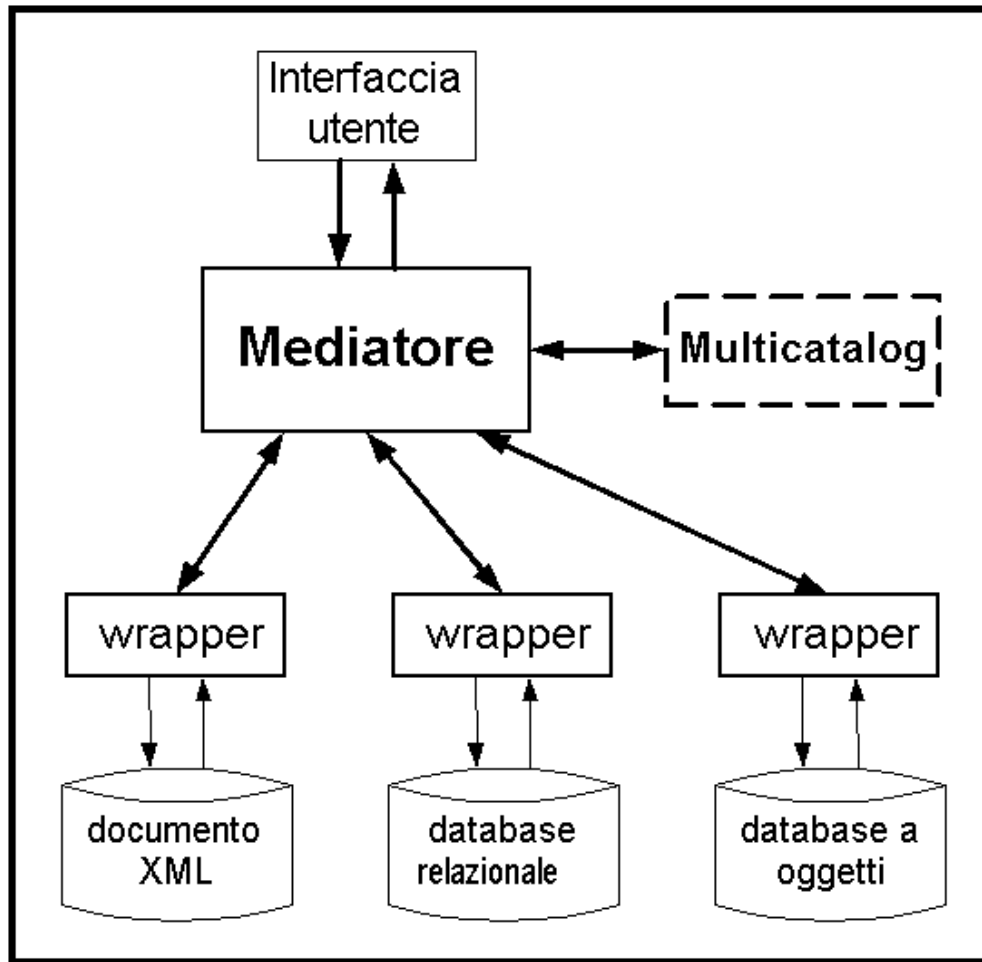


Figura 2.7: Architettura multicatalog

2.7.3 Il Multicatalog

Per mezzo di un facilitatore [13] o di un mediatore, come il sistema Momis [14], è possibile raccogliere dati da diverse fonti per realizzare un unico catalogo virtuale.

In figura 2.7 è rappresentata l'architettura per la realizzazione e la gestione di un multicatalog: nella parte in basso abbiamo le sorgenti, i singoli cataloghi delle varie ditte nei vari formati (documenti XML, database relazionali o a oggetti), alle quali il sistema si interfaccia per mezzo dei wrapper che in fase di integrazione raccolgono i dati e le loro strutture da fornire al mediatore, mentre in fase di interrogazione o di modifica passano le query ai rispettivi cataloghi; al di sopra dei wrapper vi sono il mediatore per l'integrazione dei dati e il query manager per la gestione delle operazioni da effettuare sui dati; infine troviamo le interfacce

utente, tipicamente sottoforma di pagine web, ma potenzialmente anche in altre forme per le imprese che utilizzano questo strumento in modo personale.

Questo approccio per la realizzazione del catalogo virtuale risolve i problemi che erano rimasti in sospeso nella realizzazione di un commercio elettronico semplice e completo sia per le imprese che lo implementano che per i clienti che ne usufruiscono.

Il multicatalog permette un enorme risparmio di tempo al cliente che non si perde nella navigazione dei siti e nel confronto di tutti i dati, ma accede a un solo sistema che gli fornisce le informazioni sui prodotti richiesti prelevandoli dalle varie fonti e selezionandoli in base alle specifiche dell'acquirente.

Il catalogo globale semplifica inoltre, l'attività delle aziende che consultandolo riescono a trovare velocemente i prodotti di cui necessitano e a identificare chi li vende ai prezzi più convenienti.

Capitolo 3

L'integrazione intelligente con MOMIS

3.1 Il sistema MOMIS

Il Mediator Environment for Multiple Information Sources (MOMIS) [14,26], è un sistema per l'integrazione intelligente delle informazioni, sviluppato sull'architettura I^3 , vista al capitolo precedente, nel progetto Murst ex-40% Interdata realizzato in collaborazione tra l'Università di Modena e Reggio Emilia e l'Università di Milano. Come si può osservare in figura 3.1 la struttura di MOMIS ricalca le specifiche I^3 .

Al livello più basso abbiamo i wrapper, che svolgono la funzione di interfaccia tra il mediatore e le sorgenti, che per il sistema MOMIS possono essere sia strutturate, database, che semistrutturate, come i documenti XML.

I compiti del wrapper sono essenzialmente due:

1. nella fase di integrazione: raccolgono le informazioni sulle strutture dei dati delle sorgenti e ne forniscono una descrizione nel linguaggio ODL_{I^3} [15] utilizzato dal sistema;
2. nella fase di query processing: traducono la query, inviata dal mediatore nel linguaggio OQL_{I^3} , in un linguaggio interpretabile dalla singola sorgente. Quindi operano il processo inverso per restituire al mediatore i risultati della query.

Al livello centrale, troviamo il Mediatore, il cuore dell'intero sistema.

Il mediatore di MOMIS è costituito da due moduli:

- il Global Schema Builder (GSB) che integra gli schemi delle diverse sorgenti in base alle descrizioni in ODL_{I^3} fornite dai wrapper e crea, con

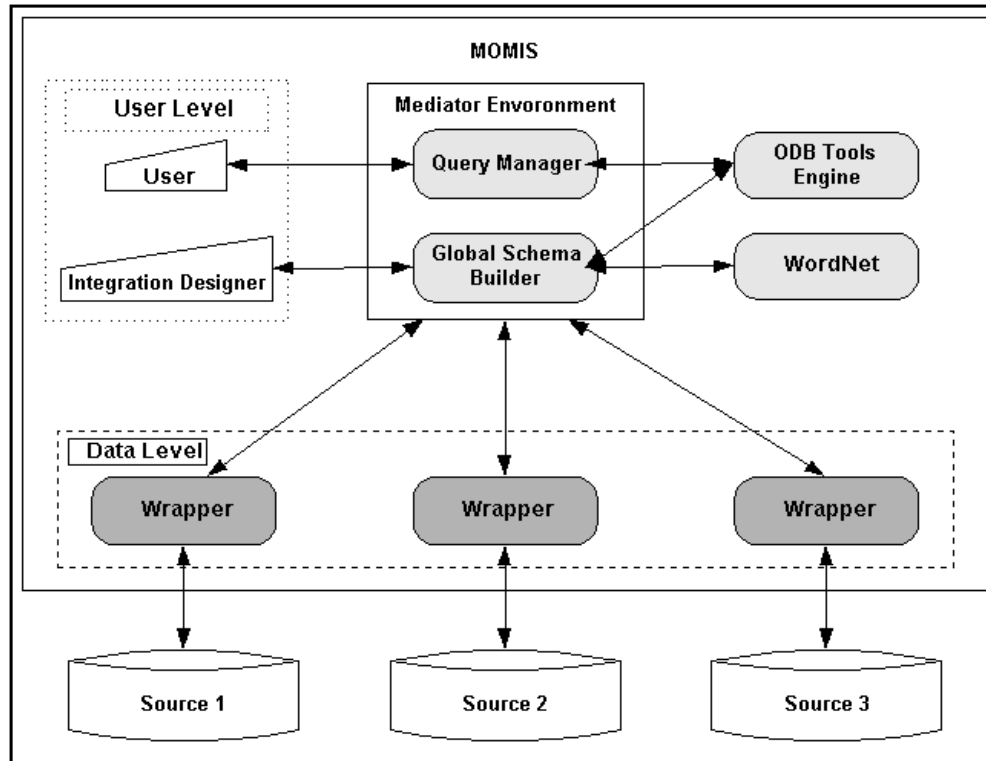


Figura 3.1: Architettura del sistema MOMIS

l'intervento del progettista, uno schema globale al quale farà riferimento nell'interazione con l'utente;

- il Query Manager [16,17] che, con l'uso di tecniche di logica descrittiva, scompone e ottimizza la query che fornirà in linguaggio OQL_{I3} ai wrapper che le applicheranno alle singole sorgenti.

L'ultimo livello che troviamo è quello in cui interviene l'utente, il quale opera interrogazioni sullo schema globale, query che passando poi per il query manager e i wrapper saranno sottoposte direttamente alle sorgenti originali.

3.1.1 Il linguaggio ODL_{I3}

Il linguaggio ODL_{I3} , realizzato in accordo con le raccomandazioni ODMG [18,19], estende le capacità del sistema ODL che già permetteva:

- definizione di tipi-classe e tipi-valore;

- distinzione fra intensione ed estensione di una classe di oggetti;
- definizione di attributi semplici e complessi;
- definizione di attributi atomici e collezioni (set, list, bag);
- definizione di relazioni binarie con relazioni inverse;
- dichiarazione della signature dei metodi.

A queste sono state aggiunte le seguenti estensioni:

- il costruttore **union**: per esprimere le strutture dati alternative nelle definizioni delle classi ODL_{T3} ;
- il costruttore **optional(*)**: per specificare se l'attributo è opzionale per l'istanza;
- le regole per l'integrità: al fine di esprimere le regole di integrità *if-then* nei livelli intra e inter-schema;
- le relazioni intensionali: che sono relazioni terminologiche di sinonimia **SYN**, ipernimia **BT**, iponimia **NT** e olonimia **RT**, che esprimono la conoscenza interschema;
- le relazioni estensionali: le relazioni SYN, BT, NT, RT estese alle classi;
- le regole di mapping: per esprimere le relazioni tra le grandezze globali e quelle locali.

3.1.2 Gli strumenti di MOMIS

Per il suo funzionamento il sistema MOMIS fa uso di due tool esterni:

- ODB tools: sia in fase di integrazione che in fase di query processing;
- WordNet: durante lo sviluppo del Thesaurus comune durante il processo di integrazione.

ODB tools [20] è un software che verifica la validità degli schemi dei database e che ottimizza semanticamente le query su basi di dati ad oggetti e si basa su due elementi:

1. l'Object Language with Complements allowing Description cycles (OL-CD): linguaggio usato per esprimere gli schemi, le query e i vincoli di integrità; dotato di tecniche di inferenza basate sulla sussunzione, tecniche tratte dalle Logiche Descrittive per l'Intelligenza Artificiale;

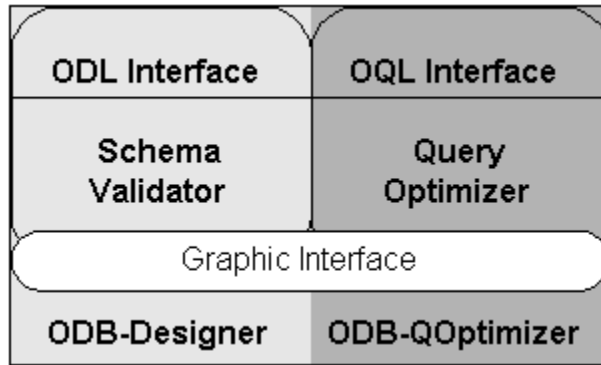


Figura 3.2: Struttura di ODB Tools

2. l'espansione semantica dei tipi, sempre attraverso l'algoritmo di sussunzione.

Come si osserva in figura 3.2 ODB tools è costituito da due parti:

ODB Designer,

ODB QOptimizer [21].

ODB Designer acquisisce e valida schemi OODB espressi in OLC, con l'algoritmo di sussunzione determina le relazioni di specializzazione tra i tipi.

ODB Optimizer serve per ottimizzare semanticamente le query, ossia in virtù delle ipotesi fatte, in [], sui vincoli di integrità che possono essere utilizzati per ottimizzare le interrogazioni, si trasforma una query in una equivalente, al limite più complessa, ma maggiormente efficiente.

Sia il processo di validazione che quello di ottimizzazione semantica sono basati sul concetto di espansione semantica di un tipo che permette di incorporare ogni possibile restrizione che non è presente nel tipo originale, ma che è logicamente implicata dallo schema.

Questo processo determina l'inserimento nello schema di nuove relazioni *isa* e quindi nuove gerarchie di ereditarietà.

Il secondo tool usato da MOMIS è **WordNet**[22,27], un database lessicale realizzato dal Cognitive Science Laboratory di Princeton.

In WordNet sono raccolti sostantivi, aggettivi, avverbi e forme verbali inglesi, organizzati per gruppi di sinonimi, detti *synset*, che rappresentano determinati concetti lessicali.

L'idea su cui è costruita la semantica lessicale è che esiste una associazione tra la forma e delle parole e il loro significato; associazione che è di tipo molti a molti e che dà origine a due proprietà:

- **sinonimia**: proprietà di un significato di avere più parole che lo possano esprimere;
- **polisemia**: proprietà di una parola di poter esprimere molteplici significati.

La corrispondenza tra parole e significati viene espressa nella così detta Matrice Lessicale, nella quale vengono mappati sulle righe i significati (cioè i synset) e sulle colonne le forme.

Se l'elemento i, j della matrice assume il valore $E_{i, j}$, non nullo, ne consegue che la forma F_j può assumere il significato M_i .

WordNet collega i termini in base a relazioni semantiche tra synset.

Le relazioni lessicali principali sono *Sinonimia*, *Antinomia*, *Ipernimia*, *Meronimia* e *Correlazione*.

La sinonimia è la relazione che stabilisce che due termini possono essere scambiati tra loro senza far mutare il significato di ciò che viene espresso.

L'antinomia è una relazione lessicale che associa due termini con significati contrari. Questa è l'unica delle relazioni considerate che si applica ai singoli termini e non ai synset.

L'ipernimia è una relazione di specializzazione tra due concetti, esprime un rapporto di tipo isa e gode delle proprietà tipiche dell'ereditarietà. Il suo inverso è l'ipernimia.

La meronimia è una relazione semantica fra due concetti x e y (x è meronimo di y) tali che x è una parte di y .

Come la relazione precedente, la meronimia gode della proprietà transitiva ed è asimmetrica, la sua relazione duale è l'olonimia.

L'ultima relazione da considerare è la correlazione che si stabilisce tra due termini che condividono lo stesso ipernimo; se ne deduce quindi che è una relazione derivabile dalle altre.

3.2 Il processo d'integrazione

La prima parte del processo di integrazione del sistema MOMIS [14], come si vede in figura 3.3, è quella che porta alla generazione del Thesaurus Comune in cui viene raccolta la conoscenza delle informazioni semantiche relative al contesto e alla struttura dei vari schemi sorgente. Terminata la costruzione del Thesaurus Comune si prosegue con il Calcolo delle affinità tra classi e termini, il cui risultato viene sfruttato dal modulo ARTEMIS [14] per la formazione dei cluster in cui vengono raggruppate le classi; cluster per ognuno dei quali nella fase conclusiva dell'integrazione viene generata una classe globale e una mapping table.

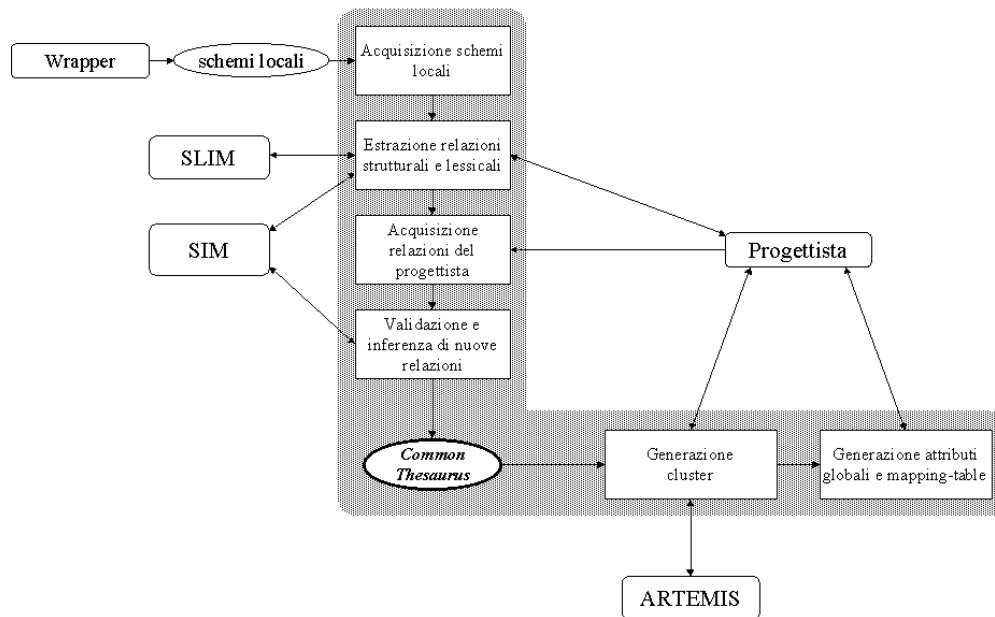


Figura 3.3: Fasi del processo di integrazione con MOMIS

3.3 Generazione del Thesaurus comune

Questa parte dell'integrazione viene operata, come si vede in figura 3.3, per mezzo dei moduli SIM, SLIM, e ARTEMIS con l'aiuto dei due tool esterni già citati. La determinazione del thesaurus si ottiene attraverso cinque passi:

1. estrazione delle relazioni intra-schema;
2. estrazione delle relazioni inter-schema;
3. arricchimento dell'insieme delle relazioni;
4. validazione delle relazioni;
5. inferenza di nuove relazioni.

3.3.1 Estrazione delle relazioni intra-schema

Una volta che il sistema ha acquisito gli schemi delle diverse sorgenti, può estrarne le gerarchie di aggregazione ed ereditarietà, in particolare quelle definite tramite foreign key.

Per una sorgente relazionale, se la foreign key è chiave primaria sia per la classe

di partenza che per quella referenziata, il modulo **SIM** trasforma la relazione tra le due classi in una relazione terminologica di tipo BT/NT, in caso contrario la caratterizza come una generica RT.

Per una sorgente a oggetti viene invece generata una relazione BT/NT per le eredità tra classi, una relazione RT per le gerarchie di aggregazione.

3.3.2 Estrazione delle relazioni inter-schema

Il secondo processo di estrazione di relazioni è quello realizzato dall'analisi e dal confronto di tutti gli schemi ODL_{T3} che portano all'identificazione di relazioni lessicali tra i nomi delle classi e degli attributi usati nei diversi schemi.

In questa fase interviene il già citato database lessicale che permette di assegnare significati appropriati alle forme base. Il compito più delicato ricade comunque sul progettista che deve cercare di specificare il maggior numero di significati, selezionandoli, però, in modo non equivoco, per evitare la generazione di relazioni errate.

Le relazioni estratte da WordNet vengono poi convertite in relazioni tipiche del Thesaurus Comune:

- la sinonimia diventa una SYN;
- l'ipernimia una BT;
- l'iponimia una NT;
- l'olonimia e la correlazione vengono invece tradotte in relazioni RT.

3.3.3 Arricchimento dell'insieme delle relazioni

In questa fase è il progettista stesso che inserisce nuove relazioni, non ricavate precedentemente, e arricchisce il Thesaurus sulla base della sua personale conoscenza.

Il passaggio è importante perchè le relazioni che vengono aggiunte devono essere corrette per evitare di generare uno schema errato.

3.3.4 Validazione delle relazioni

Una volta estratte le relazioni intra e inter-schema, queste devono essere validate per evitare, in fase di integrazione, l'uso di relazioni errate.

Questo compito come già accennato viene affidato al sistema ODB tools che analizza la compatibilità tra i domini degli attributi che partecipano a una relazione, e nel caso questi verifichino i criteri di validità, li valida e li inserisce a tutti gli

effetti nel Thesaurus Comune.

I criteri di validità sono fondamentalmente tre:

1. una relazione SYN è valida se i domini dei due attributi che vi prendono parte sono equivalenti, oppure se sono l'uno la specializzazione dell'altro;
2. una relazione BT è corretta se il dominio del primo attributo contiene o è equivalente a quello del secondo;
3. quando il dominio di un attributo è definito utilizzando il costruttore *union* la relazione che coinvolge quell'attributo è corretta se i criteri già citati sono rispettati da almeno uno dei suoi domini.

3.3.5 Inferenza di nuove relazioni

Anche questa parte del processo di generazione del Thesaurus Comune, come la precedente, ricade su ODB tools che con tecniche di sussunzione derivate dall'intelligenza artificiale inferisce nuove relazioni.

Il sistema compie un'operazione di scomposizione e riorganizzazione degli schemi locali per individuare nuove relazioni tra le varie classi che vengono collegate in questa struttura provvisoria per mezzo delle relazioni già esistenti:

- ogni BT/NT dà luogo a una gerarchia di ereditarietà;
- una SYN genera una doppia gerarchia di ereditarietà;
- una RT produce una aggregazione.

Sulla base di queste informazioni viene creata una struttura di collegamenti tra le classi dalla quale ODB tools desume nuove relazioni di aggregazione ed ereditarietà, concludendo il processo di arricchimento del Thesaurus Comune.

3.4 Il calcolo delle affinità

Una volta conclusa la formazione del Thesaurus Comune, per proseguire nell'integrazione è necessario calcolare l'affinità tra le classi locali per poterle raggruppare nei cluster che porteranno alla formazione degli schemi globali.

Questo calcolo viene realizzato, dal modulo ARTEMIS, con l'uso di due parametri:

1. lo *Structural Affinity Coefficient* [14] che viene determinato tra due classi, in base alle relazione tra i loro attributi;
2. il *Name Affinity Coefficient* [14] che si basa sulle relazione che legano le coppie di classi.

La combinazione di questi due coefficienti produce il *Global Affinity Coefficient* [14], il vero elemento di confronto per stabilire la similarità tra due classi.

Per un raffronto numerico delle affinità tra i termini viene assegnato un peso $\sigma_{\mathfrak{R}}$ a ogni relazione, peso che sarà tanto maggiore quanto più stringente sarà il legame imposto dalla relazione ai due termini considerati:

$$\sigma_{syn} \geq \sigma_{bt/nt} \geq \sigma_{rt}$$

Tipicamente Vengono considerati i seguenti valori:

$$\begin{aligned}\sigma_{syn} &= 1; \\ \sigma_{bt} &= \sigma_{nt} = 0.8; \\ \sigma_{rt} &= 0.5.\end{aligned}$$

3.5 La generazione dei cluster

Calcolato il grado di affinità tra le classi, tutte quelle che hanno una affinità superiore a una soglia prestabilita vengono riunite in gruppo dal mediatore, secondo tecniche di clustering [24].

La procedura di clustering è iterativa e inizia allocando un cluster per ogni classe, quindi ad ogni successiva iterazione fonde i cluster delle due classi che hanno il Global Affinity Coefficient massimo. La procedura ha termine quando tutte le classi appartengono a un unico cluster.

Il risultato più interessante di questo processo non è però il cluster finale, ma l'albero che viene generato, in cui le foglie sono le classi da raggruppare.

In quest'albero ogni nodo rappresenta un livello di clusterizzazione ed è caratterizzato da un coefficiente di affinità tra i due sottoalberi che collega.

Come detto, la formazione dei cluster avviene confrontando i coefficienti di affinità dei nodi con un valore soglia, tipicamente viene usato 0,5; tutte le classi dei sottoalberi che hanno un grado di affinità superiore alla soglia vengono fuse in un unico cluster, ossia in un'unica classe globale.

3.6 La costruzione delle classi globali

Una volta che sono stati realizzati i cluster in cui sono state raccolte le classi locali, si passa alla fase di integrazione vera e propria degli schemi.

Per ogni cluster realizzato si crea una classe globale caratterizzata da:

- un nome che fa da identificatore;
- un insieme di attributi;

- una mapping table che gestisce la corrispondenza tra gli attributi globali e i dati delle sorgenti locali.

Dopo l'implementazione delle classi globali, il processo di integrazione prosegue con altre due fasi: la fusione degli attributi e la creazione della mapping table. Poiché le regole di mapping definite nella seconda fase dipendono dalle fusioni operate nella prima, il sistema MOMIS ne tiene traccia operando i due passi finali contemporaneamente.

Per comprenderne meglio il funzionamento verranno, però, analizzati separatamente.

3.7 Fusione degli attributi

La fusione degli attributi nasce dall'esigenza di eliminare le ridondanze e di integrare completamente gli schemi.

La modalità di fusione degli attributi all'interno di una classe globale è dipendente dal tipo di relazioni che li legano, e dal fatto che queste siano o meno validate.

3.7.1 Fusione degli attributi di relazioni validate

Come già detto, i modi di fondere gli attributi sono funzione dei tipi di relazione che li legano, vediamo quindi caso per caso.

Consideriamo due attributi di classi locali differenti, legati da una relazione di sinonimia (SYN), questo significa che esprimono lo stesso significato e quindi possono essere fusi.

A questo punto del processo il mediatore analizza i domini dei diversi attributi: se coincidono li fonde nell'attributo globale il cui dominio contiene quello di tutti gli altri; se sono diversi, propone un elenco di domini, che generalizzano quelli degli attributi locali, dal quale il progettista dovrà scegliere quello da utilizzare.

Per quanto riguarda gli attributi legati da relazioni di specializzazione BT/NT, il sistema procede fondendo l'attributo che sta più in basso, nella gerarchia di generalizzazione, in quello che sta più in alto.

Il comportamento riguardo ai domini è il medesimo descritto per le relazioni di sinonimia.

Le relazioni di associazione RT non vengono considerate in questa fase, poiché non esprimono legami forti gestibili dal sistema in modo automatico, il loro utilizzo, per la fusione degli attributi, viene lasciato al progettista.

Un caso particolare di fusione si verifica quando attributi legati da relazioni di sinonimia, compaiono anche in relazioni di specializzazione.

Il sistema è realizzato in modo tale da risolvere comunque questo problema in

Mapping table	Attributo globale 1	Attributo globale 2	Attributo globale 3	Attributo globale 4
Schema locale 1	Attributo locale 1.1	Attributo locale 1.2	Attributo locale 1.3	Attributo locale 1.4
Schema locale 2	Attributo locale 2.1	Attributo locale 2.2	Attributo locale 2.3	Attributo locale 2.4
Schema locale 3	Attributo locale 3.1	Attributo locale 3.2	Attributo locale 3.3	Attributo locale 3.4

Figura 3.4: Esempio di mapping table

automatico, dando priorità alle relazioni più stringenti, quindi fondendo prima gli attributi legati da SYN.

Ovviamente al termine del processo automatico il progettista può intervenire e modificare lo stato generato dal sistema.

3.7.2 Fusione di attributi di relazioni non validate

Per gli attributi legati da relazioni non validate il sistema non riesce a identificarne la compatibilità e quindi non può operarne la fusione in modo automatico.

Questo caso ricade quindi sulla conoscenza del progettista che si farà carico delle fusioni e delle scelte dei domini degli attributi globali creati.

3.8 La generazione delle mapping table

Le mapping table sono le tabelle degli schemi globali che conservano le informazioni necessarie per passare dagli attributi globali a quelli locali, cioè ai dati veri e propri. Sono evidentemente uno strumento fondamentale soprattutto in fase di query processing. Come si può osservare in figura 3.4 si ha una riga per ogni classe locale e una colonna per ogni attributo globale; e gli elementi della tabella sono gli attributi locali che sono stati mappati in quelli globali.

Le mapping rule [14] che legano gli attributi globali ai valori degli attributi locali possono essere cinque:

1. **corrispondenza semplice:** l'attributo globale assume banalmente il valore dell'attributo locale;

2. **corrispondenza in AND:** questo caso si può avere quando due attributi di una stessa classe locale sono stati fusi insieme, quindi si avrà che l'attributo globale assumerà i valori concatenati degli attributi locali considerati;
3. **corrispondenza in UNION:** è analoga alla precedente con la differenza che i valori degli attributi locali sono in alternativa, cioè l'attributo globale assumerà solo il valore di uno o dell'altro attributo locale;
4. **valore di default:** è una costante assegnata dal progettista per valorizzare informazioni presenti negli schemi sottoforma di metadato, ma che non sono state prese in considerazione dal sistema;
5. **valore NULL:** serve chiaramente per identificare la mancanza di corrispondenza tra un attributo globale e gli attributi di un determinato schema locale.

Capitolo 4

SI-Designer: il tool per la creazione dello schema globale

4.1 SI-Designer in MOMIS

SI-Designer [23] è l'Interfaccia Utente Grafica (GUI) del sistema MOMIS; è a questa che accede il progettista per l'integrazione intelligente delle sorgenti e la costruzione delle classi globali.

Come si vede in figura 4.1 SI-Designer è un contenitore per gli altri moduli, visti al capitolo tre, necessari alla costruzione dello schema globale:

- SIM (Source Integrator Module)
- SLIM (Sources Lexical Integrator Module)
- ARTEMIS (Analysis and Reconciliation Tool Environment for Multiple Information Sources)
- TUNIM [25] (TUNing of the Mapping table)

La raffinazione delle informazioni, fatta nel passaggio attraverso questi moduli di SI-Designer porta all'integrazione finale delle sorgenti locali.

Vediamo quindi come si usa SI-Designer.

4.2 L'automa di utilizzo di SI-Designer

Prima di entrare nel dettaglio dei vari elementi presenti nella GUI di MOMIS vediamo come si possono susseguire le fasi di utilizzo dei vari moduli. Si inizia col pannello Source nel quale si caricano i wrapper per acquisire le informazioni sulle sorgenti, quindi si passa al pannello Thes.Rel in cui il modulo SIMA estrae

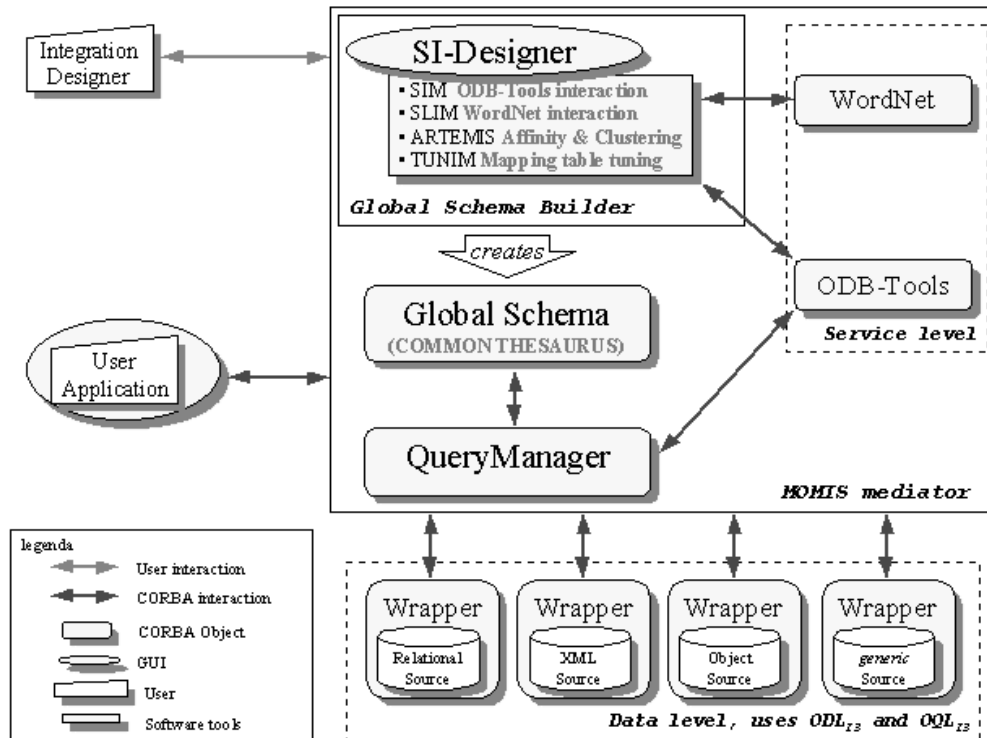


Figura 4.1: Architettura del sistema MOMIS

le relazioni intra-schema. Fissate queste prime relazioni si passa al pannello di SLIM che per mezzo del modulo WordNet consente la costruzione di nuove relazioni lessicali tra i termini che caratterizzano classi e attributi. Si torna quindi al pannello Thes.rel dove il progettista può intervenire aggiungendo o rimuovendo delle relazioni. Al termine di queste operazioni il progettista richiama SIMB che per mezzo di ODB tools inferisce nuove relazioni e controlla e valida le relazioni di tutto il Thesaurus. Validate le operazioni si passa al pannello Cluster dove per mezzo del modulo ARTEMIS le classi locali vengono raggruppate in cluster; cluster che determineranno le mapping table. Nel pannello TUNIM si modificano le mapping table fondendo gli attributi per rimuovere le ridondanze. Completata questa operazione si giunge alla formazione dello schema globale.

Come si può vedere dalle linee tratteggiate dell'automata è permesso al progettista di tornare indietro, nel processo di integrazione, per operare delle revisioni e delle correzioni:

- può ritornare a SLIM per modificare il significato o la forma base di un termine per il quale erano state generate relazioni non corrette;

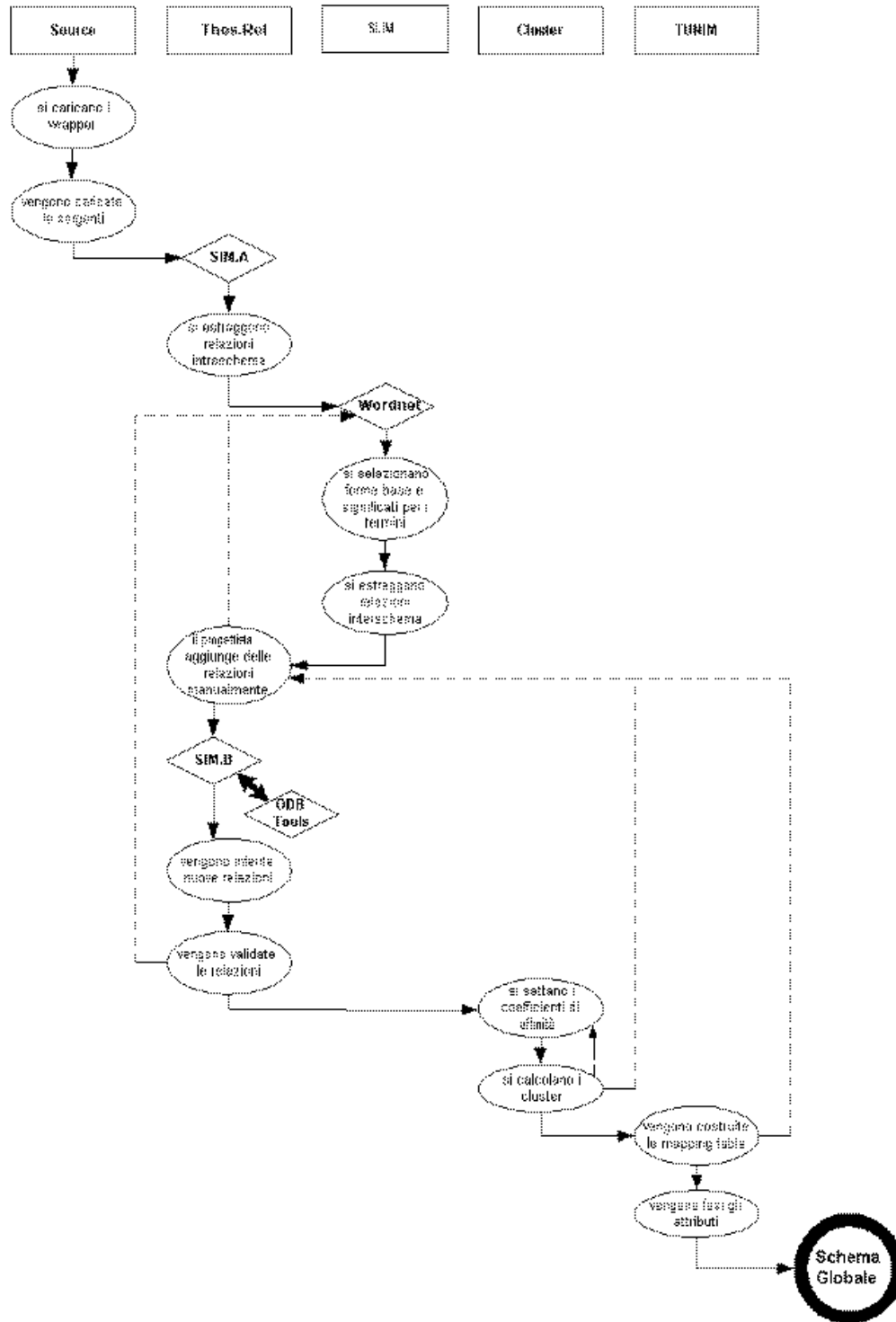


Figura 4.2: Automa di funzionamento di SI-Designer

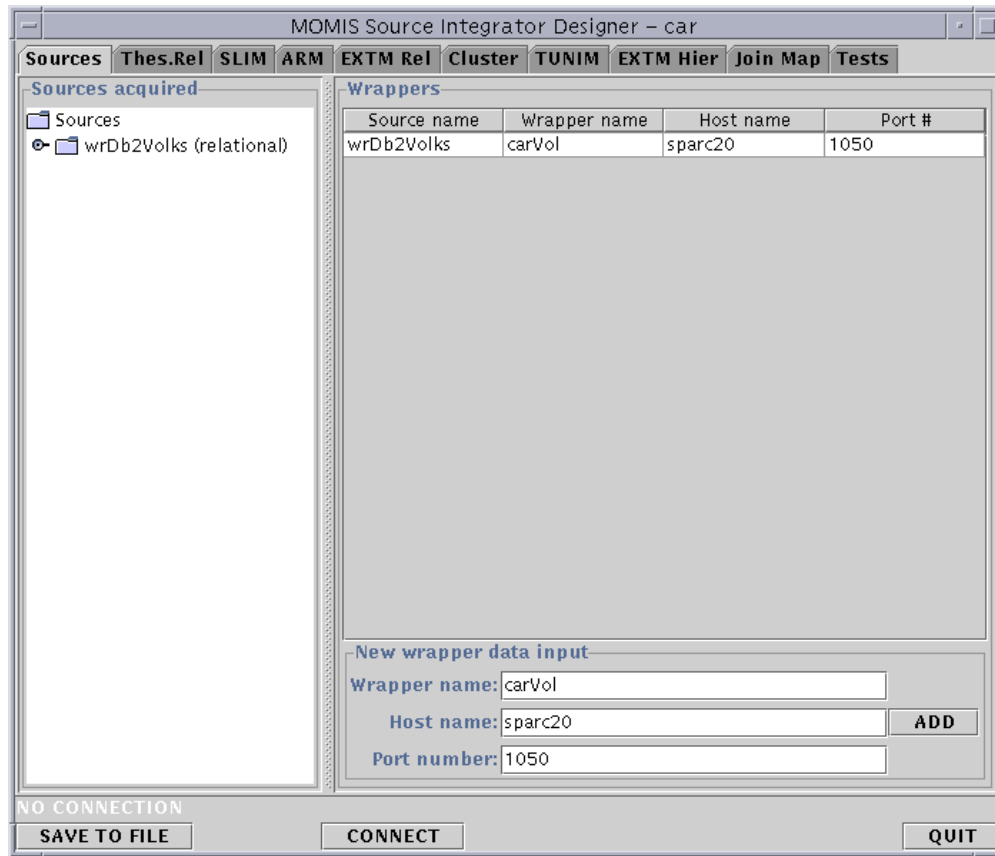


Figura 4.3: Pannello Source

- può tornare a Thes.Rel e inserire relazioni che si accorge, a posteriori, essere state trascurate;
- è possibile fare un passo indietro nel pannello cluster per modificare i parametri di ARTEMIS poichè erano stati generati cluster non corretti o poco soddisfacenti.

4.3 Source

Il primo passo da compiere nell' integrazione è, come abbiamo visto nell' automa, il caricamento dei wrapper e quindi delle informazioni sulle sorgenti.

La prima finestra che ci presenta SI-Designer è quindi il pannello *Source*, che possiamo vedere in figura 4.3. La parte in basso di questo pannello, denominata

textitNew wrapper data input, è quella necessaria al caricamento dei wrapper; qui bisogna inserire:

- il nome del wrapper;
- la macchina su cui il wrapper gira;
- il numero della porta attraverso cui vi si può accedere.

Inseriti i dati e cliccando sul tasto ADD si carica il wrapper.

Questa operazione produce la comparsa dei dati relativi al wrapper e alla sorgente a cui si interfaccia nella finestra di label textitWrappers e la visualizzazione della struttura della sorgente, con la specificazione del tipo (relazionale, a oggetti, semistrutturata), nella finestra di destra denominata textitSource acquired.

4.4 SIM A

Una volta caricate le sorgenti si prosegue nel processo di integrazione con la formazione del Thesaurus Comune il cui primo passo consiste nell'estrazione delle relazioni intra-schema; per fare questo si accede al pannello *Thes.Rel*, figura 4.4, e si seleziona il pulsante run SIMA.

Il risultato di questa operazione sarà la comparsa nella finestra di destra delle relazioni tra le classi e gli attributi, di una stessa sorgente, dovute alla presenza delle foreign key negli schemi.

Nella finestra di destra si possono osservare i due termini che partecipano alla relazione: *Source*, *Destination* separati dal tipo di relazione che li lega: SYN, BT, NT RT. Questi elementi sono seguiti da una colonna in cui viene identificato, per mezzo di un colore e di un codice, chi ha costruito la relazione. Per terminare sulla destra si può osservare una colonna che segnala se la relazione è stata o meno validata.

4.5 SLIM

L'operazione successiva all'estrazione delle relazioni intra-schema consiste nella costruzione delle relazioni lessicali tra i termini degli attributi e delle classi dei diversi schemi, questo si compie nel pannello *SLIM*.

Vediamo quindi come si fa ad assegnare un significato a un termine:

- per prima cosa si seleziona col tasto sinistro del mouse il termine che interessa,

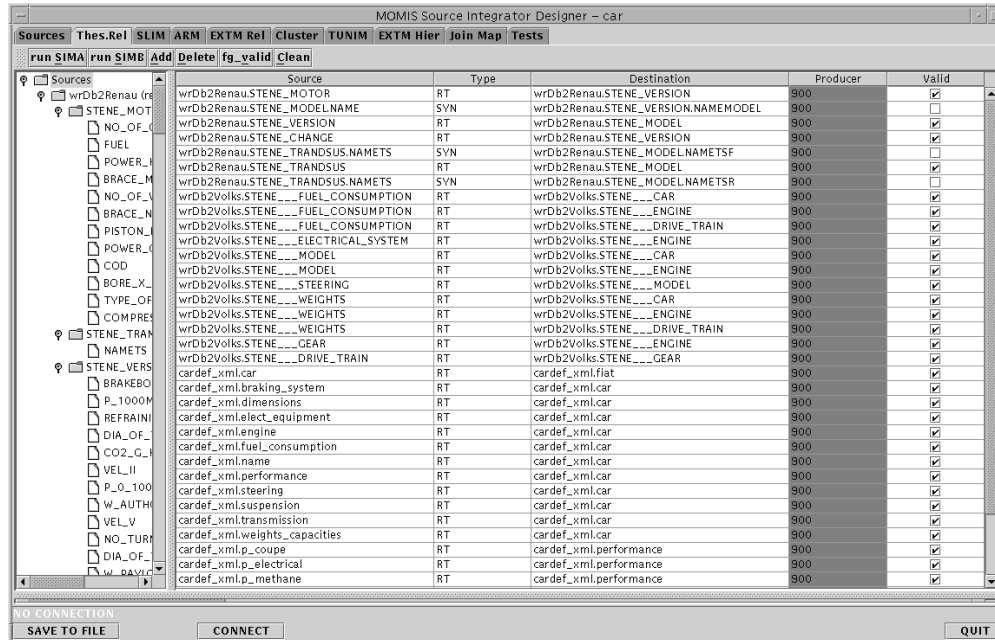


Figura 4.4: Pannello Thes.Rel: run SIM

- quindi vi si clicca sopra col tasto destro, questo provoca la comparsa di una tendina, figura 4.5, con i seguenti elementi che conducono alla selezione del significato del termine:
 - word form: selezionando questa voce si apre una finestra in cui si può inserire la forma base per il termine considerato;
 - select sense: questo elemento causa l'apertura di una finestra contenente i significati di WordNet relativi alla forma base scelta per quel termine. In questa finestra è possibile selezionare il significato o i significati più adeguati;
 - ignored term: questa voce permette di obbligare SLIM a non considerare il termine. Questa operazione può essere necessaria, per esempio, quando la forma base che si assegna al termine avrebbe un significato eccessivamente generale che produrrebbe la formazione di molte relazioni lessicali, la maggior parte delle quali sarebbe sbagliata;
 - Hypernym: selezionando questo elemento il progettista può scegliere il significato del termine selezionando l'ambito a cui il significato deve appartenere e quindi la specificazione più corretta in questo ambito (figura 4.6).

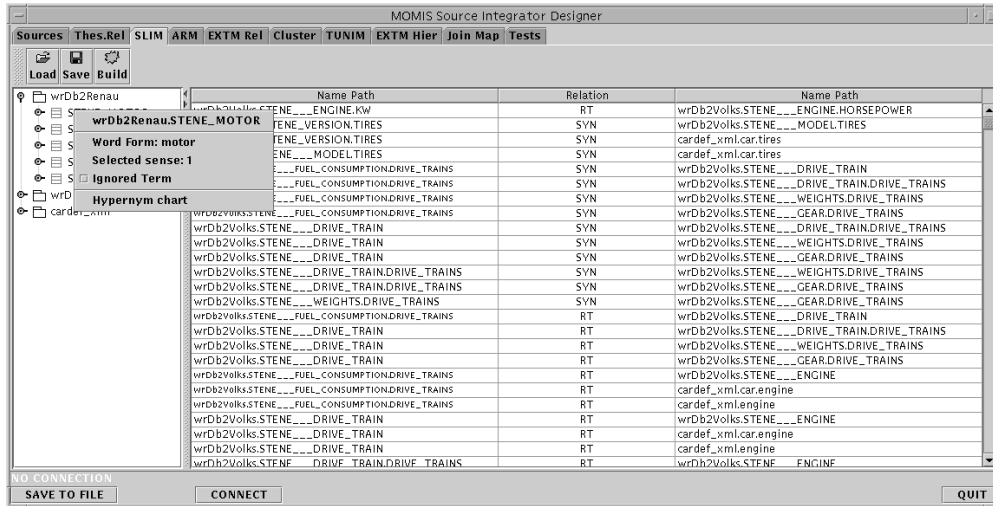


Figura 4.5: Pannello SLIM

- quando per tutti i termini è stata compiuta questa operazione si clicca sul tasto BUILD che spinge SLIM alla costruzione di tutte le relazioni lessicali che riesce a desumere in base ai significati scelti per i vari termini.

Poichè le sorgenti hanno spesso un lungo elenco di termini e quindi occorre molto tempo per lo svolgimento di questa operazione sono stati revisti due tasti, LOAD e SAVE, per caricare e salvare il significati già assegnati ai termini. In alternativa al pannello SLIM si può utilizzare il pannello ARM, che svolge le medesime funzioni, solamente con una interfaccia a pannelli invece che a menu.

4.6 SIM B

Costruite le relazioni dal modulo lessicale si ritorna al pannello Thes.Rel, qui il progettista può inserire nuove relazioni frutto della sua personale conoscenza, per mezzo del tasto ADD che fa comparire una finestra in cui inserire il termine source, quello destination e la relazione che li lega.

Inoltre per mezzo del tasto DELETE il progettista può rimuovere delle relazioni, che ritiene errate, dal Thesurus.

Una volta concluse queste operazioni di arricchimento, modifica e correzione delle relazioni, il progettista preme il tasto run SIMB che fa compiere al sistema il processo di inferenza di nuove relazioni e il controllo e la validazione di tutte le relazioni contenute nel Thesaurus.

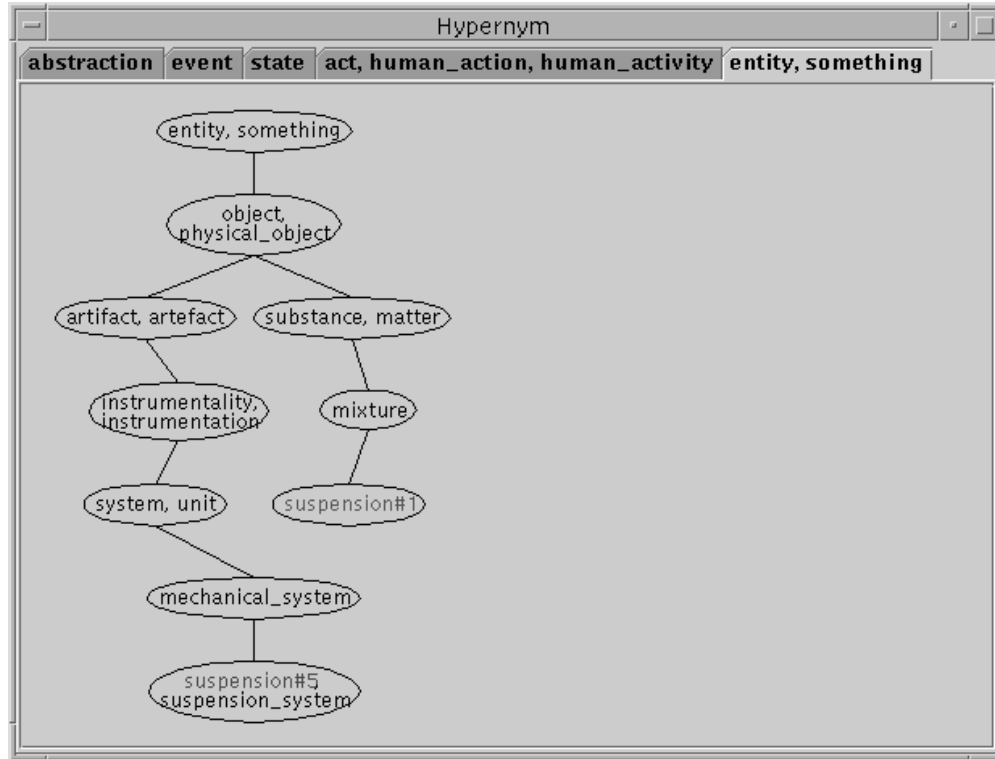


Figura 4.6: Pannello Hypernym

4.7 EXTM Rel

Questo pannello, figura 4.7, permette al progettista di sfruttare la sua cono-scenza sui legami estensionali tra le classi in modo da costruire nuove relazioni per il Thesurus.

Nella parte destra del pannello, denominata *New rule data input*, si possono selezionare tramite i menu a tendina le due classi di cui si vuole esprimere il legame, quindi si può scegliere il tipo di relazione estensionale che sussiste tra i due tra i due termini proposti dal menu a tendina, centrale: EQU, ISA.

Una volta stabilito se le due classi sono una la specializzazione dell' altra o se sono coincidenti, si può premere il tasto ADD che provoca l' inserimento della relazione estensionale e la sua conversione in relazione intensionale rispettivamente di tipo SYN o NT, relazione che viene poi inserita nel Thesaurus.

Per la descrizione teorica\ sistemistica si veda [28].

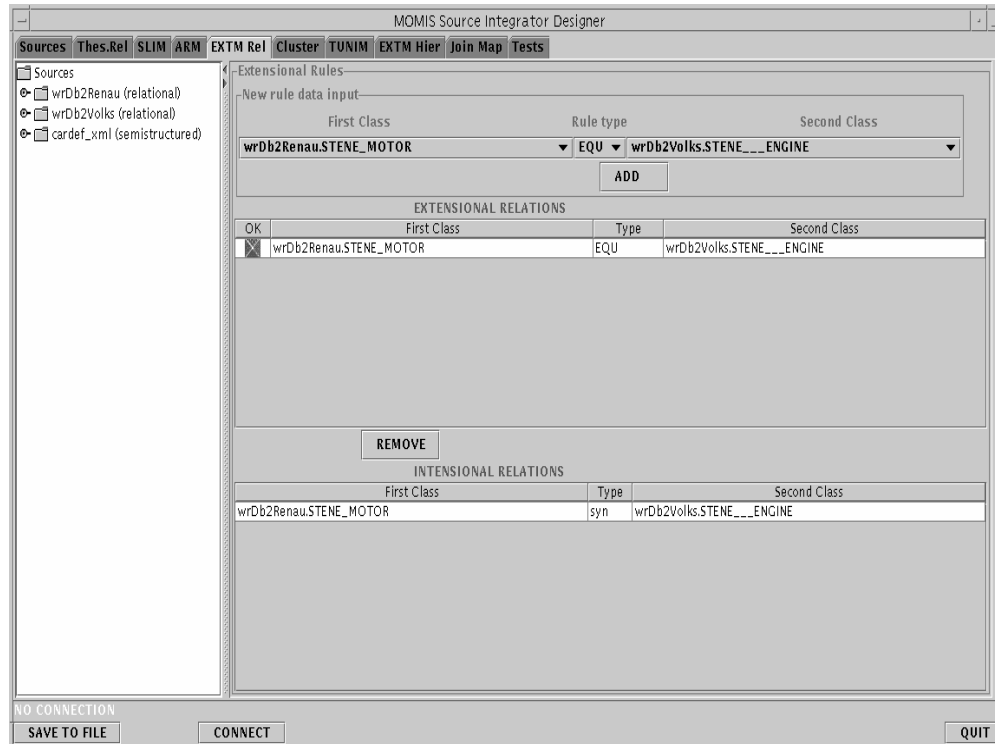


Figura 4.7: Pannello EXT M Rel

4.8 Cluster

Il pannello Cluster è l'interfaccia grafica del modulo ARTEMIS e serve, come già detto, alla costruzione dei cluster, nei quali vengono raggruppate le classi locali dei vari schemi che devono essere integrati.

Questo pannello presenta due sheets:

- *Cluster Tuning;*
- *ARTEMIS Configuration.*

Nel pannello *ARTEMIS Configuration*, visibile in figura 4.8, si configura il modulo ARTEMIS settando:

- i pesi delle relazioni del Thesaurus;
- i valori di soglia per le affinità e il clustering;
- i coefficienti per l'affinità globale.

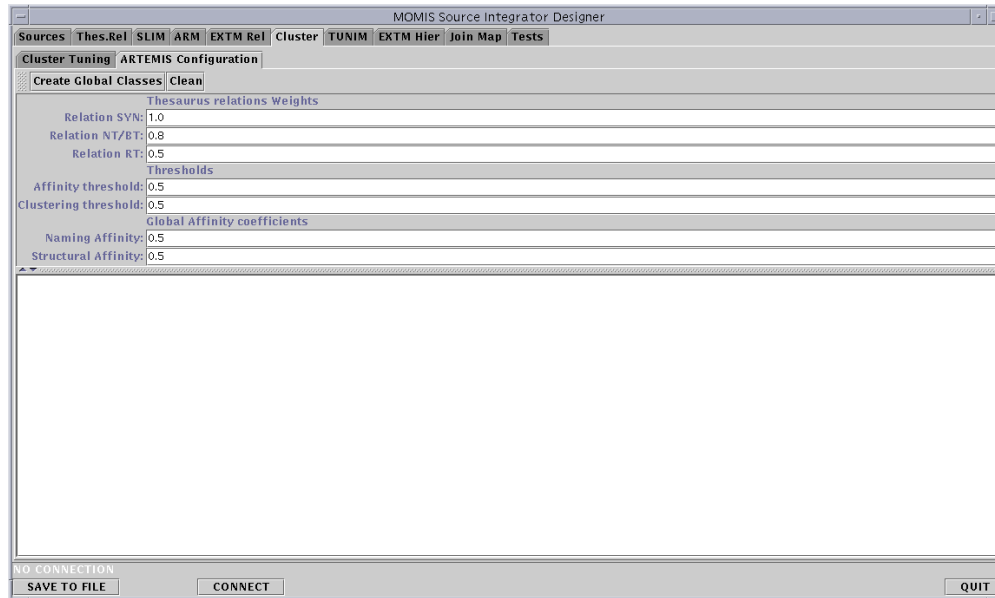


Figura 4.8: Pannello Cluster: ARTEMIS Configuration

Una volta settate le variabili di ARTEMIS si procede premendo il pulsante CREATE GLOBAL CLASSES che fa eseguire dal sistema il modulo ARTEMIS e fornisce nella finestra in basso i cluster costruiti secondo le specifiche.

Se non si vogliono modificare i valori standard dei parametri di ARTEMIS si può eseguire la costruzione dei cluster anche dal pannello Cluster Tuning (figura 4.9) invocando il modulo per mezzo del pulsante CREATE GLOBAL CLASSES USING ARTEMIS.

Il risultato viene presentato nella finestra di sinistra. Su questo risultato il progettista può intervenire con i pulsanti che si trovano al centro, per una modifica manuale.

Selezionando una classe e premendo il pulsante Rename Class è possibile cambiare il nome a una classe globale; questa è una operazione consueta poichè i cluster creati non hanno nomi significativi.

Con il pulsante Delete Mapping è possibile cancellare la mappa costruita.

I pulsanti Delete Class e Add new Class servono per rimuovere o aggiungere una classe globale.

Con i pulsanti UnMap Interf. e Map Interface il progettista può rimuovere o mappare un attributo in una classe globale selezionata.

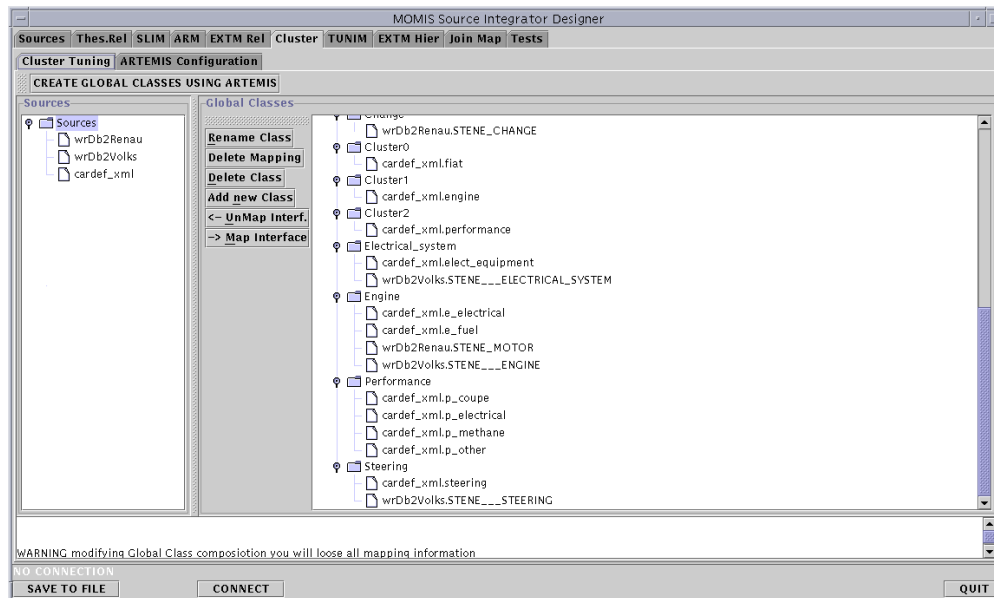


Figura 4.9: Pannello Cluster: Cluster Tuning

4.9 TUNIM

Il pannello TUNIM, che possiamo osservare in figura 4.10, è quello che permette la costruzione degli schemi globali dopo che le classi locali sono state raggruppate in cluster dal modulo ARTEMIS.

La prima operazione da compiere è premere il tasto MAP ALL che provoca la creazione di un attributo globale per ogni attributo locale, nelle rispettive classi globali determinate dai cluster.

Successivamente per mezzo del tasto FUSE ATTR si compie la fusione degli attributi locali uguali in un unico attributo globale rimuovendo così le ridondanze.

Terminate queste operazioni automatiche il progettista può intervenire manualmente per modificare lo schema, da qui la presenza di altri pulsanti nel pannello:

- map: produce la mappatura dell'elemento scelto nella finestra *Attribute not mapped*, nell'attributo globale selezionato;
- unmap: rimuove un attributo locale dalla mappatura e lo pone nella finestra *Attribute not mapped*;
- relation: crea una finestra in cui visualizza le relazioni a cui partecipa l'attributo locale selezionato tra quelli non mappati;

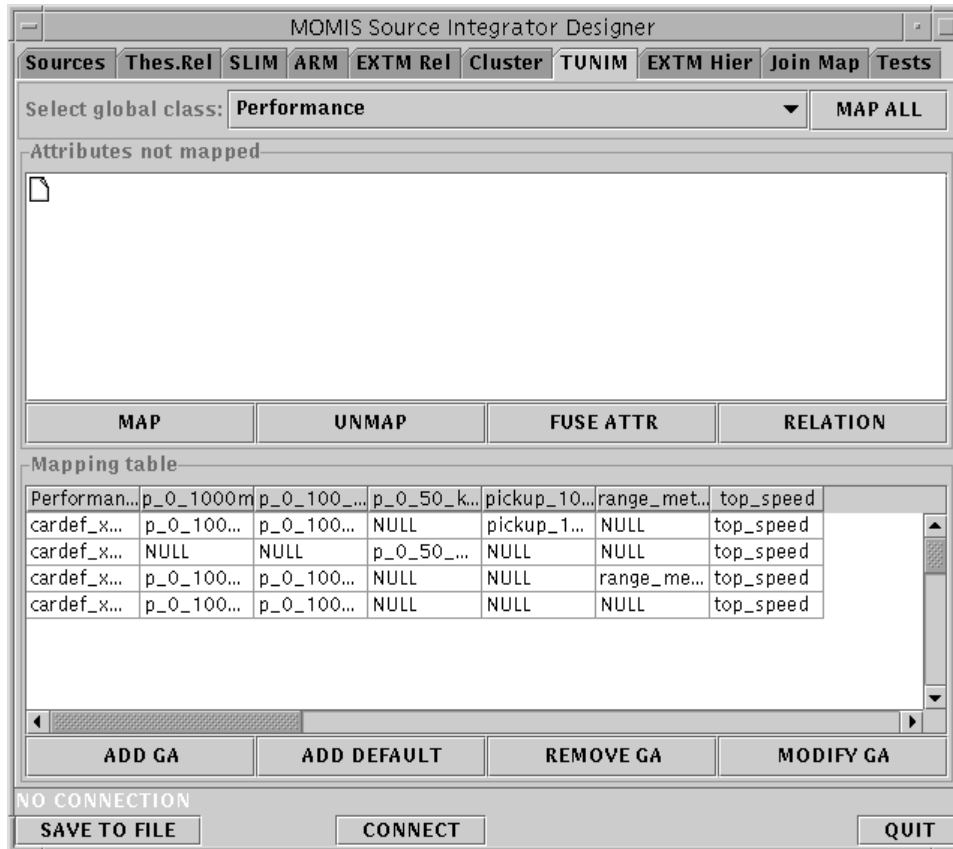


Figura 4.10: Pannello TUNIM

- add ga: permette l'inserimento di un nuovo attributo globale, per il quale il progettista dovrà scegliere nome e dominio;
- add default: consente di assegnare un valore di default a un determinato attributo locale;
- remove ga: rimuove l'attributo globale selezionato dalla mapping table. Se l'attributo non è vuoto, ma in esso vi sono mappati degli attributi locali, questi vengono spostati nella finestra *Attribute not mapped*;
- modify ga: consente di modificare nome e domini di un attributo globale.

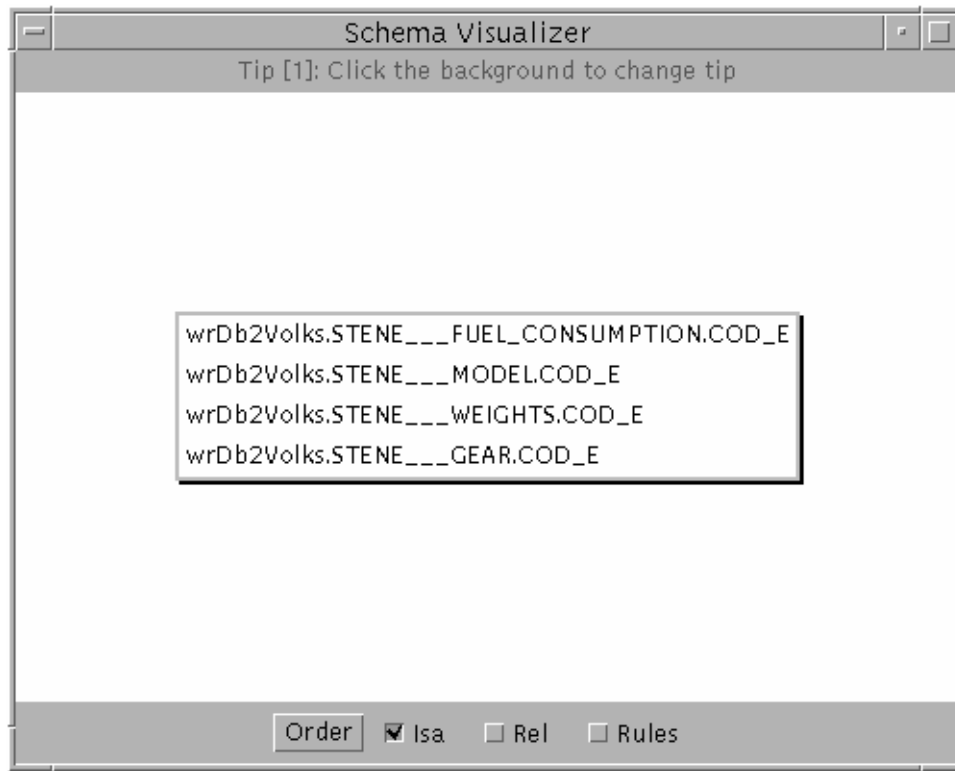


Figura 4.11: Pannello TUNIM: Schema Visualizer

Capitolo 5

Il catalogo virtuale realizzato con il sistema MOMIS

5.1 Obiettivi

L'obiettivo di questa tesi è presentare come sia possibile realizzare un catalogo virtuale per il commercio elettronico che risolva i problemi evidenziati al capitolo due. Con l'uso del sistema MOMIS per l'integrazione intelligente delle informazioni si è costruito un catalogo globale combinando i dati estratti da più cataloghi di singole imprese. L'esempio generato è evidentemente più finalizzato a un commercio elettronico di tipo B2C, ma i risultati, come già evidenziato al capitolo due sono facilmente applicabili anche a un e-commerce B2B.

L'esempio implementato è finalizzato a un commercio elettronico per la vendita di automobili; per renderlo il più reale possibile, in modo tale da testare anche le concrete capacità del sistema MOMIS, si sono raccolti i dati delle auto di tre importanti case automobilistiche:

- Renault;
- Fiat;
- Volkswagen.

I dati sono relativi al mese di giugno 2000 e sono stati prelevati dalle versioni inglesi dei siti web delle rispettive case. L'esigenza dei dati in lingua inglese deriva dal fatto che il modulo WordNet utilizzato dal sistema MOMIS è per ora disponibile solo in questa versione.

5.2 Le fonti

Il sistema MOMIS prevede per il processo di integrazione di raccogliere dati da fonti strutturate (database) o semistrutturate (come documenti XML), la prima fase del progetto è stata dunque quella di trasformazione dei dati, presentati in modo non strutturato nelle pagine HTML dei rispettivi rivenditori, in un formato utilizzabile nel processo di integrazione. Al fine di rendere significativo e non banale il processo di integrazione, in modo da evidenziarne anche le possibili lacune, si è cercato di differenziare il più possibile le strutture iniziali. I database relazionali sono stati realizzati con DB2 il DBMS di IBM, mentre per la sorgente semistrutturata si è usato il linguaggio XML.

5.2.1 Volkswagen

La prima fonte di informazioni analizzata è stato il catalogo web della Volkswagen. Per la discreta regolarità mostrata nella presentazione dei dati dei diversi modelli di automobili si è deciso di ricostruirla per il nostro esempio come fonte strutturata e in particolare come un database relazionale.

Dall'analisi dei dati e dai collegamenti tra attributi e modelli si è determinato, per costruire la base di dati, lo schema E/R di figura 5.1, in cui si possono osservare le varie classi. Di seguito possiamo vedere come sono stati mappati gli attributi delle automobili all'interno delle classi dello schema, e di che tipi vengono dichiarati in DB2:

CAR

```
NAME CHAR(15) NOT NULL
WHEELBASE DECIMAL(6,2) NOT NULL
FRONT_TRACK DECIMAL(6,2) NOT NULL
REAR_TRACK DECIMAL(6,2) NOT NULL
LENGTH DECIMAL(6,2) NOT NULL
WIDTH DECIMAL(6,2) NOT NULL
HEIGHT DECIMAL(6,2) NOT NULL
GROUND_CLEARANCE DECIMAL(6,2) NOT NULL
PRIMARY KEY (NAME)
```

DRIVE_TRAIN

```
DRIVE_TRAINS CHAR(50) NOT NULL
```

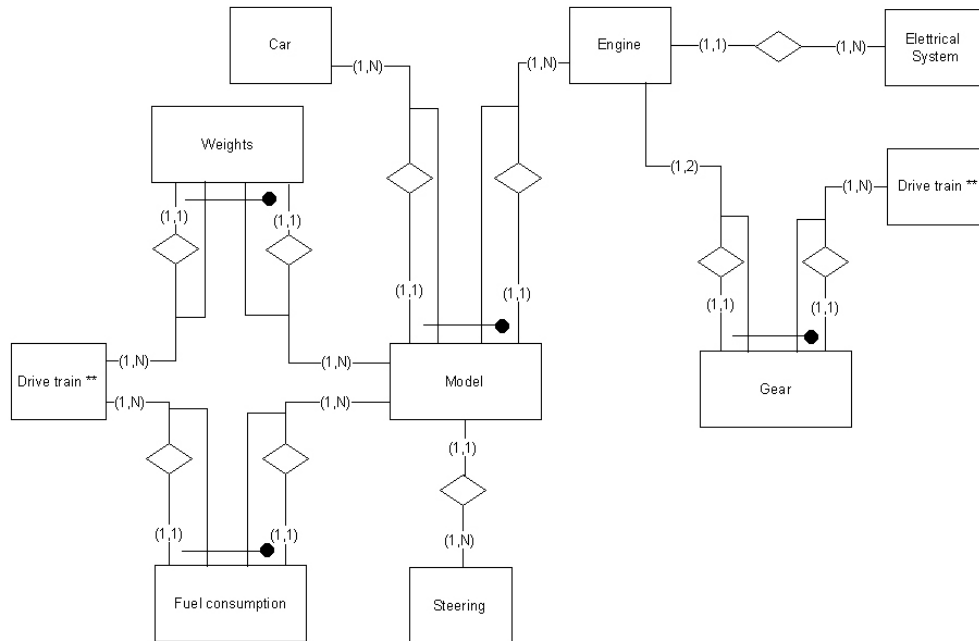


Figura 5.1: Diagramma E\R del database Volkswagen

```
TRANS_GEAR_RATIO CHAR(20) NOT NULL
PRIMARY KEY (DRIVE_TRAINS,TRANS_GEAR_RATIO)
```

```
ELECTRICAL_SYSTEM
```

```
ALTERNATOR_V_A CHAR(20) NOT NULL
BATTERY_V INTEGER NOT NULL
BATTERY_A_HRS INTEGER NOT NULL
IGNITION CHAR(70) NOT NULL
PRIMARY KEY (IGNITION,ALTERNATOR_V_A,BATTERY_V, BATTERY_A_HRS)
```

```
ENGINE
```

```
COD_E CHAR(15) NOT NULL
TYPE CHAR(80) NOT NULL
BORE_CM DECIMAL(5,2) NOT NULL
STROKE_CM DECIMAL(5,2) NOT NULL
```


DISPLACEMENT_CC INTEGER NOT NULL
COMPRESSION_RATIO CHAR(10) NOT NULL
HORSEPOWER CHAR(15) NOT NULL
kW CHAR(15) NOT NULL
MAX_TORQUE_MKG CHAR(15) NOT NULL
MAX_TORQUE_NM CHAR(15) NOT NULL
FUEL_REQUIREMENT CHAR(25) NOT NULL
ARRANGEMENT CHAR(50) NOT NULL
CYLINDER_BLOCK CHAR(50) NOT NULL
CRANCK_SHAFT CHAR(50) NOT NULL
CYLINDER_HEAD CHAR(50) NOT NULL
VALVE_TRAIN CHAR(200) NOT NULL
COOLING_SYSTEM CHAR(200) NOT NULL
LUBRIFICATION CHAR(200) NOT NULL
FUEL_AIR_SUPPLY CHAR(100) NOT NULL
EMISSION CHAR(220) NOT NULL
ALTERNATOR_V_A CHAR(20) NOT NULL
BATTERY_V INTEGER NOT NULL
BATTERY_A_HRS INTEGER NOT NULL
IGNITION CHAR(50) NOT NULL
FIRE_ORDER CHAR(15) NOT NULL
ENGINE_OIL_L DECIMAL(5,2) NOT NULL
FUEL_TANK_L DECIMAL(5,2) NOT NULL
COOLING_SYSTEM_L DECIMAL(5,2) NOT NULL
WIPER_FLUID_L DECIMAL(5,2) NOT NULL
PRIMARY KEY (COD_E)
FOREIGN KEY (IGNITION,ALTERNATOR_V_A,BATTERY_V,BATTERY_A_HRS)
REFERENCES ELECTRICAL_SYSTEM (IGNITION,
ALTERNATOR_V_A,BATTERY_V,BATTERY_A_HRS)

GEAR

COD_E CHAR(15) NOT NULL
DRIVE_TRAINS CHAR(50) NOT NULL
TRANS_GEAR_RATIO CHAR(20) NOT NULL
FIRST CHAR(10) NOT NULL
SECOND CHAR(10) NOT NULL
THIRD CHAR(10) NOT NULL
FOURTH CHAR(10) NOT NULL
FIFTH CHAR(10)

REVERSE CHAR(10)
FINAL_DRIVE CHAR(10)
PRIMARY KEY (COD_E,DRIVE_TRAINS,TRANS_GEAR_RATIO)
FOREIGN KEY (COD_E) REFERENCES ENGINE(COD_E)
FOREIGN KEY (DRIVE_TRAINS,TRANS_GEAR_RATIO)
REFERENCES DRIVE_TRAIN(DRIVE_TRAINS,TRANS_GEAR_RATIO)

STEERING

STEERING CHAR(35) NOT NULL
PRIMARY KEY (STEERING)

MODEL

NAME CHAR(15) NOT NULL
COD_E CHAR(15) NOT NULL
EPA_CLASS CHAR(20) NOT NULL
PASSENGER_VOL_DM3 INTEGER NOT NULL
TRUNCK_VOL_DM3 INTEGER NOT NULL
SEATING_CAPACITY INTEGER NOT NULL
VOL_FRONT_DM3 INTEGER NOT NULL
VOL_REAR_DM3 INTEGER NOT NULL
HEAD_ROOM_FRONT_CM DECIMAL(5,1) NOT NULL
HEAD_ROOM_REAR_CM DECIMAL(5,1) NOT NULL
SHOULDER_ROOM_FRONT_CM DECIMAL(5,1) NOT NULL
SHOULDER_ROOM_REAR_CM DECIMAL(5,1) NOT NULL
LEG_ROOM_FRONT_CM DECIMAL(5,1) NOT NULL
LEG_ROOM_REAR_CM DECIMAL(5,1) NOT NULL
BODY_TYPE CHAR(80) NOT NULL
SUSPENSION_FRONT CHAR(200) NOT NULL
SUSPENSION_REAR CHAR(200) NOT NULL
SERVICE_BRAKE CHAR(200) NOT NULL
ANTI_LOCK_BRAK_SYS CHAR(80) NOT NULL
PARKING_BRAKE CHAR(80) NOT NULL
WHEELS CHAR(100) NOT NULL
TIRES CHAR(100) NOT NULL
DRAG_COEFFICIENT DECIMAL(5,2) NOT NULL
STEERING CHAR(50) NOT NULL
TURNS_LOCK2LOCK DECIMAL(5,2) NOT NULL

TURNING_CIRCLE_C2C_M DECIMAL(5,2) NOT NULL
RATIO CHAR(10) NOT NULL
PRICE DECIMAL(8,2) NOT NULL
PRIMARY KEY (NAME,COD_E)
FOREIGN KEY (COD_E) REFERENCES ENGINE(COD_E)
FOREIGN KEY (NAME) REFERENCES CAR(NAME)
FOREIGN KEY (STEERING) REFERENCES STEERING(STEERING)

FUEL_CONSUMPTION

NAME CHAR(15) NOT NULL
COD_E CHAR(15) NOT NULL
DRIVE_TRAINS CHAR(50) NOT NULL
TRANS_GEAR_RATIO CHAR(20) NOT NULL
CITY_KM_L INTEGER NOT NULL
HIGHWAY_KM_L INTEGER NOT NULL
PRIMARY KEY (NAME,COD_E,DRIVE_TRAINS,TRANS_GEAR_RATIO)
FOREIGN KEY (NAME) REFERENCES CAR(NAME)
FOREIGN KEY (COD_E) REFERENCES ENGINE(COD_E)
FOREIGN KEY (DRIVE_TRAINS,TRANS_GEAR_RATIO)
REFERENCES DRIVE_TRAIN(DRIVE_TRAINS,TRANS_GEAR_RATIO)

WEIGHTS

NAME CHAR(15) NOT NULL
COD_E CHAR(15) NOT NULL
DRIVE_TRAINS CHAR(50) NOT NULL
TRANS_GEAR_RATIO CHAR(20) NOT NULL
CURB_KG INTEGER
PAYLOAD_KG INTEGER
PRIMARY KEY (NAME,COD_E,DRIVE_TRAINS,TRANS_GEAR_RATIO)
FOREIGN KEY (NAME) REFERENCES CAR(NAME)
FOREIGN KEY (DRIVE_TRAINS,TRANS_GEAR_RATIO)
REFERENCES DRIVE_TRAIN(DRIVE_TRAINS,TRANS_GEAR_RATIO)
FOREIGN KEY (COD_E) REFERENCES ENGINE(COD_E)

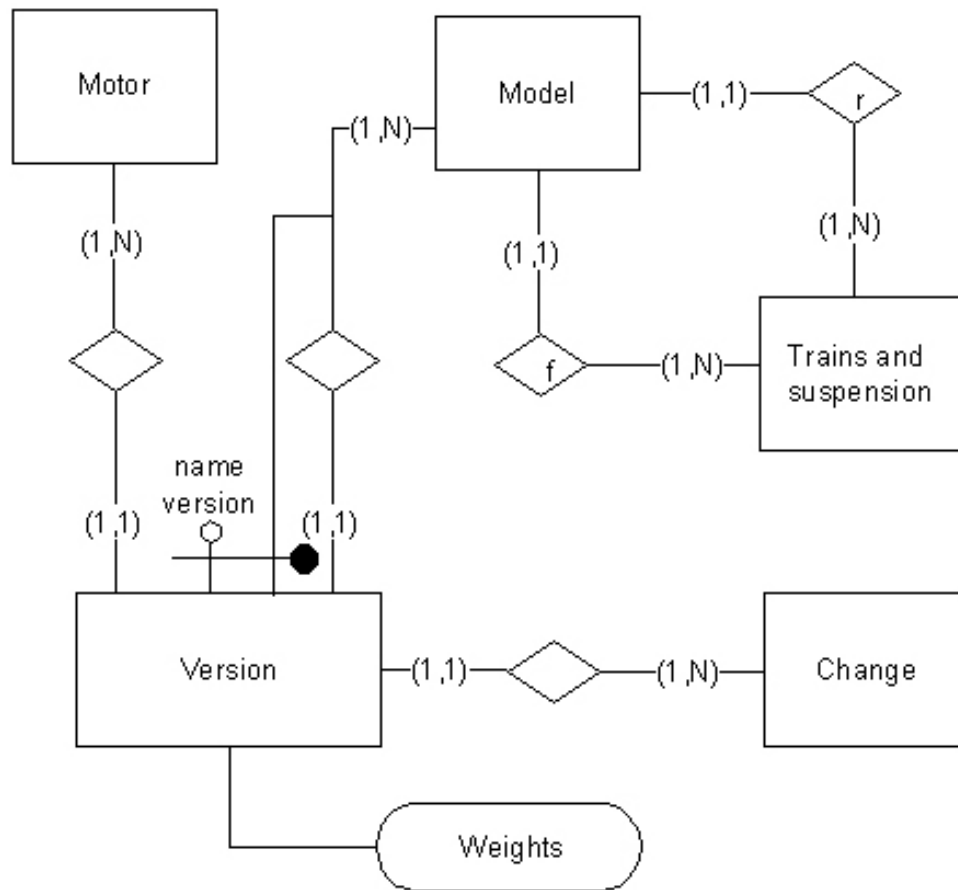


Figura 5.2: Diagramma E\R del databse Renault

5.2.2 Renault

La seconda sorgente da integrare è stata ricostruita per i modelli di automobili della Renault.

Benchè ci fosse un maggiore variabilità di attributi tra i vari modelli, rispetto a quanto avveniva per la Volkswagen, appariva ancora una struttura di fondo abbastanza regolare, per questo anche questa sorgente è stata realizzata come database. Come per il caso precedente possiamo osservare in figura 5.2 lo schema E/R e qui di seguito le classi e gli attributi caratterizzati dai rispettivi tipi:

TRANDSUS

NAMETS CHAR(200) NOT NULL

PRIMARY KEY (NAMETS)

CHANGE

CODC CHAR(5) NOT NULL
TYPE CHAR(10) NOT NULL
PRIMARY KEY (CODC,TYPE)

MODEL

NAME CHAR(15) NOT NULL
NAMETSF CHAR(200) NOT NULL
NAMETSR CHAR(200) NOT NULL
VOLBAG_DM3 INTEGER NOT NULL
LENGTH DECIMAL(8,4) NOT NULL
WIDTH DECIMAL(8,4) NOT NULL
HEIGHT DECIMAL(8,4) NOT NULL
PRIMARY KEY (NAME)
FOREIGN KEY (NAMETSF) REFERENCES TRANDSUS(NAMETS)
FOREIGN KEY (NAMETSR) REFERENCES TRANDSUS(NAMETS)

MOTOR

COD CHAR(5) NOT NULL
PISTON_DISPLACEMENT_CC INTEGER NOT NULL
BORE_X_RACE_MM CHAR(10) NOT NULL
NO_OF_CYLINDER INTEGER NOT NULL
NO_OF_VALVES INTEGER NOT NULL
COMPRESSION_RATIO CHAR(8) NOT NULL
POWER_KW CHAR(15) NOT NULL
POWER_CV CHAR(15) NOT NULL
BRACE_NM CHAR(15) NOT NULL
BRACE_MKG CHAR(15) NOT NULL
TYPE_OF_INJECTION CHAR(50) NOT NULL
FUEL CHAR(15) NOT NULL
PRIMARY KEY (COD,TYPE_OF_INJECTION)

VERSION

COD CHAR(5) NOT NULL
NAMEMODEL CHAR(15) NOT NULL
TYPE_OF_INJECTION CHAR(50) NOT NULL
NAMEVERSION CHAR(30) NOT NULL
CODC CHAR(5) NOT NULL
TYPECHANGE CHAR(10) NOT NULL
NO_OF_RELATIONSHIP INTEGER NOT NULL
VEL_I DECIMAL(8,3) NOT NULL
VEL_II DECIMAL(8,3) NOT NULL
VEL_III DECIMAL(8,3) NOT NULL
VEL_IV DECIMAL(8,3)
VEL_V DECIMAL(8,3)
WHEELS CHAR(10) NOT NULL
TIRES CHAR(10) NOT NULL
ABILITY_TANK_L INTEGER
AERODYNAMICS CHAR(10)
DIA_OF_TURN_SIDEWALK DECIMAL(8,2) NOT NULL
DIA_OF_TURN_WALL DECIMAL(8,2) NOT NULL
NO_TURN_FLYING DECIMAL(8,2)
SPEED_MAX INTEGER NOT NULL
P_0_100_S DECIMAL(8,2) NOT NULL
P_400M_S DECIMAL(8,2)
P_1000M_S DECIMAL(8,2)
CITY_CICLE DECIMAL(6,2) NOT NULL
CITY_EXTRA_CICLE DECIMAL(6,2) NOT NULL
MIXED_CICLE DECIMAL(6,2) NOT NULL
CO2_G_KM INTEGER
REFRAINING_CIRCUIT CHAR(3) NOT NULL
ABS CHAR(20)
BRAKEBOOSTER CHAR(10) NOT NULL
DIA_BRAKEBOOSTER INTEGER
FRONT CHAR(15) NOT NULL
REAR CHAR(15) NOT NULL
W_EMPTY_MARCH_ORDER_KG INTEGER NOT NULL
W_AUTHORIZED_MAX_KG INTEGER NOT NULL
W_TRAVELLING_TOTAL_KG INTEGER NOT NULL
W_PAYLOAD_KG INTEGER NOT NULL
W_REFRAINED_TOWING_KG INTEGER NOT NULL
W_NOT_REFRAINED_TOWING_KG INTEGER NOT NULL

```

PRICE DECIMAL(8,2)
PRIMARY KEY (NAMEMODEL,NAMEVERSION)
FOREIGN KEY (COD,TYPE_OF_INJECTION)
REFERENCES MOTOR (COD,TYPE_OF_INJECTION)
FOREIGN KEY (NAMEMODEL) REFERENCES MODEL (NAME)
FOREIGN KEY (CODC, TYPECHANGE) REFERENCES CHANGE (CODC,TYPE)

```

5.2.3 Fiat

L'ultimo catalogo considerato è stato quello della Fiat. In questo caso si è riscontrata una notevole variabilità negli attributi che descrivevano i diversi modelli, motivo per cui risultava difficile una ricostruzione strutturata per un database, si è quindi deciso di implementarla come sorgente semistrutturata in un documento XML. Per prima cosa si è realizzato il DTD per mantenere le informazioni sulla struttura del documento, quindi nel rispetto delle regole del DTD si sono inseriti i dati delle auto della Fiat del documento XML. Vediamo il codice del DTD:

```

<!ELEMENT fiat (car*)>

<!ELEMENT car (name, engine, transmission, steering,
braking_system, suspension, dimensions, tires, elect_equipment,
weights_capacities, performance, fuel_consumption, price)>
<!ELEMENT name (#PCDATA)>
<!ATTLIST name
id ID #REQUIRED>
<!ELEMENT tires (#PCDATA)>
<!ELEMENT price (#PCDATA)>
<!ELEMENT engine (e_fuel | e_electrical)>

<!ELEMENT e_fuel (e_name, cylinders?, layout?, bore_mm?,
capacity_cc?, compression_ratio?, power_kw, power_bhp,
torque_nm, torque_kgm, valve_gear?, fuel_system?, ignition?)>
<!ELEMENT e_name (#PCDATA)>
<!ELEMENT cylinders (#PCDATA)>
<!ELEMENT layout (#PCDATA)>
<!ELEMENT bore_mm (#PCDATA)>
<!ELEMENT capacity_cc (#PCDATA)>
<!ELEMENT compression_ratio (#PCDATA)>
<!ELEMENT power_kw (#PCDATA)>
<!ELEMENT power_bhp (#PCDATA)>
<!ELEMENT torque_nm (#PCDATA)>

```

```
<!ELEMENT torque_kgm (#PCDATA)>
<!ELEMENT valve_gear (#PCDATA)>
<!ELEMENT fuel_system (#PCDATA)>
<!ELEMENT ignition (#PCDATA)>

<!ELEMENT e_electrical (e_name, e_type, layout, power_kw,
torque_nm, engine_control, drive_battery_kwh, bat_charge_time)>
<!ELEMENT e_type (#PCDATA)>
<!ELEMENT engine_control (#PCDATA)>
<!ELEMENT drive_battery_kwh (#PCDATA)>
<!ELEMENT bat_charge_time (#PCDATA)>

<!ELEMENT transmission (drive, speed)>
<!ELEMENT drive (#PCDATA)>
<!ELEMENT speed (#PCDATA | gear)*>
<!ELEMENT gear (g1, g2, g3, g4, g5?, g6?, rm, final_drive_ratio?)>
<!ELEMENT g1 (#PCDATA)>
<!ELEMENT g2 (#PCDATA)>
<!ELEMENT g3 (#PCDATA)>
<!ELEMENT g4 (#PCDATA)>
<!ELEMENT g5 (#PCDATA)>
<!ELEMENT g6 (#PCDATA)>
<!ELEMENT rm (#PCDATA)>
<!ELEMENT final_drive_ratio (#PCDATA)>

<!ELEMENT steering (s_type, turning_circle)>
<!ELEMENT s_type (#PCDATA)>
<!ELEMENT turning_circle (#PCDATA)>

<!ELEMENT braking_system (b_front_mm, b_rear_mm)>
<!ELEMENT b_front_mm (#PCDATA)>
<!ELEMENT b_rear_mm (#PCDATA)>

<!ELEMENT suspension (s_front, s_rear)>
<!ELEMENT s_front (#PCDATA)>
<!ELEMENT s_rear (#PCDATA)>

<!ELEMENT dimensions (length_m, width_m, height_m,
luggage_capacity_dm3, wheelbase_m?, track_front_mm?,
track_rear_mm?)>
<!ELEMENT length_m (#PCDATA)>
```



```
<!ELEMENT width_m (#PCDATA)>
<!ELEMENT height_m (#PCDATA)>
<!ELEMENT luggage_capacity_dm3 (#PCDATA)>
<!ELEMENT wheelbase_m (#PCDATA)>
<!ELEMENT track_front_mm (#PCDATA)>
<!ELEMENT track_rear_mm (#PCDATA)>

<!ELEMENT elect_equipment (battery_capacity,
alternator_current?)>
<!ELEMENT battery_capacity (#PCDATA)>
<!ELEMENT alternator_current (#PCDATA)>

<!ELEMENT weights_capacities (fuel_cap_l, kerb_weight_kg,
towable_weight_kg?)>
<!ELEMENT fuel_cap_l (#PCDATA)>
<!ELEMENT kerb_weight_kg (#PCDATA)>
<!ELEMENT towable_weight_kg (#PCDATA)>

<!ELEMENT performance (p_electrical | p_methane
| p_coupe | p_other)>
<!ELEMENT p_other (top_speed, p_0_100_km_h?, p_0_1000m?)>
<!ELEMENT top_speed (#PCDATA)>
<!ELEMENT p_0_100_km_h (#PCDATA)>
<!ELEMENT p_0_1000m (#PCDATA)>
<!ELEMENT p_electrical (top_speed, p_0_50_km_h)>
<!ELEMENT p_0_50_km_h (#PCDATA)>
<!ELEMENT p_methane (top_speed, p_0_100_km_h, p_0_1000m,
range_methane_km)>
<!ELEMENT range_methane_km (#PCDATA)>
<!ELEMENT p_coupe (top_speed, p_0_100_km_h, p_0_1000m,
pickup_1000m_sec)>
<!ELEMENT pickup_1000m_sec (#PCDATA)>
<!ELEMENT fuel_consumption (urban, outoftown?, combined,
urban_range_km?)>
<!ELEMENT urban (#PCDATA)>
<!ELEMENT outoftown (#PCDATA)>
<!ELEMENT combined (#PCDATA)>
<!ELEMENT urban_range_km (#PCDATA)>
```

Vediamo anche un esempio per meglio comprendere la descrizione in XML:

```
<fiat>
  <car>
    <name id="a">Barchetta 1.8 16V</name>
    <engine>
      <e_fuel>
        <e_name>1,8 16v</e_name>
        <cylinders>4</cylinders>
        <layout>front transverse</layout>
        <bore_mm>82 x 82,7</bore_mm>
        <capacity_cc>1747</capacity_cc>
        <compression_ratio>10,3:1</compression_ratio>
        <power_kw>96@6300</power_kw>
        <power_bhp>130@6300</power_bhp>
        <torque_nm>164@4300</torque_nm>
        <torque_kgm>16,7@4300</torque_kgm>
        <valve_gear>2 OHC (toothed belt) with variable valve
        timing and hydraulic tappets</valve_gear>
        <fuel_system>phased sequential electronic MPI
        </fuel_system>
        <ignition>electronic, static advance combined with
        injection</ignition>
      </e_fuel>
    </engine>
    <transmission>
      <drive>front</drive>
      <speed>5</speed>
    </transmission>
    <steering>
      <s_type>Rack and pinion with power steering</s_type>
      <turning_circle>10,5</turning_circle>
    </steering>
    <braking_system>
      <b_front_mm>257</b_front_mm>
      <b_rear_mm>240</b_rear_mm>
    </braking_system>
    <suspension>
      <s_front>independent MacPherson with lower wishbones
      and coil springs</s_front>
      <s_rear>independent with coil springs and lower
```

```

wishbones</s_rear>
</suspension>
<dimensions>
  <length_m>3,91</length_m>
  <width_m>1,64</width_m>
  <height_m>1,26</height_m>
  <luggage_capacity_dm3>165</luggage_capacity_dm3>
</dimensions>
<tires>195/55 R 15</tires>
<elect_equipment>
  <battery_capacity>50</battery_capacity>
</elect_equipment>
<weights_capacities>
  <fuel_cap_l>50</fuel_cap_l>
  <kerb_weight_kg>1060</kerb_weight_kg>
</weights_capacities>
<performance>
  <p_other>
    <top_speed>200</top_speed>
    <p_0_100_km_h>8,9</p_0_100_km_h>
  </p_other>
</performance>
<fuel_consumption>
  <urban>11,6</urban>
  <outoftown>6,5</outoftown>
  <combined>8,4</combined>
</fuel_consumption>
<price>19621</price>
</car>
</fiat>

```

5.3 I wrapper

Seguendo il processo di integrazione descritto al capitolo tre, il primo passo da fare, una volta selezionate le sorgenti da integrare, è quello di attivare i wrapper che acquisiscono le strutture degli schemi delle sorgenti, e le traducono nel linguaggio usato da MOMIS, cioè in ODL_{I^3} . Vediamo quindi come risultano le strutture una volta tradotte:

Volkswagen

```
// Data source Description... for []
// metadata
// ----- [STENE.CAR]
interface STENE__CAR(source relational wrDb2Volks
key (NAME)) {
// ResultSet [Index Info]

attribute char NAME;
attribute long /* DECIMAL */ WHEELBASE;
attribute long /* DECIMAL */ FRONT_TRACK;
attribute long /* DECIMAL */ REAR_TRACK;
attribute long /* DECIMAL */ LENGTH;
attribute long /* DECIMAL */ WIDTH;
attribute long /* DECIMAL */ HEIGHT;
attribute long /* DECIMAL */ GROUND_CLEARANCE;
};

// ----- [STENE.DRIVE_TRAIN]
interface STENE__DRIVE_TRAIN(source relational wrDb2Volks
key (DRIVE_TRAINS,TRANS_GEAR_RATIO)) {
// ResultSet [Index Info]

attribute char DRIVE_TRAINS;
attribute char TRANS_GEAR_RATIO;
};

// ----- [STENE.ELECTRICAL_SYSTEM]
interface STENE__ELECTRICAL_SYSTEM(source relational
wrDb2Volks
key (ALTERNATOR_V_A,BATTERY_V,BATTERY_A_HRS,
IGNITION)) {
// ResultSet [Index Info]

attribute char ALTERNATOR_V_A;
attribute long /* INTEGER */ BATTERY_V;
attribute long /* INTEGER */ BATTERY_A_HRS;
attribute char IGNITION;
};

// ----- [STENE.ENGINE]
interface STENE__ENGINE(source relational wrDb2Volks
```

```

key (COD_E)
foreign_key(ALTERNATOR_V_A,BATTERY_V,BATTERY_A_HRS,IGNITION)
references STENE___ELECTRICAL_SYSTEM) {
// ResultSet [Index Info]

attribute char COD_E;
attribute char TYPE;
attribute long /* DECIMAL */ BORE_CM;
attribute long /* DECIMAL */ STROKE_CM;
attribute long /* INTEGER */ DISPLACEMENT_CC;
attribute char COMPRESSION_RATIO;
attribute char HORSEPOWER;
attribute char KW;
attribute char MAX_TORQUE_MKG;
attribute char MAX_TORQUE_NM;
attribute char FUEL_REQUIREMENT;
attribute char ARRANGEMENT;
attribute char CYLINDER_BLOCK;
attribute char CRANCK_SHAFT;
attribute char CYLINDER_HEAD;
attribute char VALVE_TRAIN;
attribute char COOLING_SYSTEM;
attribute char LUBRIFICATION;
attribute char FUEL_AIR_SUPPLY;
attribute char EMISSION;
attribute char ALTERNATOR_V_A;
attribute long /* INTEGER */ BATTERY_V;
attribute long /* INTEGER */ BATTERY_A_HRS;
attribute char IGNITION;
attribute char FIRE_ORDER;
attribute long /* DECIMAL */ ENGINE_OIL_L;
attribute long /* DECIMAL */ FUEL_TANK_L;
attribute long /* DECIMAL */ COOLING_SYSTEM_L;
attribute long /* DECIMAL */ WIPER_FLUID_L;
};

// ----- [STENE.FUEL_CONSUMPTION]
interface STENE___FUEL_CONSUMPTION(source relational wrDb2Volks
key (NAME,COD_E,DRIVE_TRAINS,TRANS_GEAR_RATIO)
foreign_key (NAME) references STENE___CAR
foreign_key (COD_E) references STENE___ENGINE

```

```

foreign_key (DRIVE_TRAINS,TRANS_GEAR_RATIO)
references STENE__DRIVE_TRAIN) {
// ResultSet [Index Info]

attribute char NAME;
attribute char COD_E;
attribute char DRIVE_TRAINS;
attribute char TRANS_GEAR_RATIO;
attribute long /* INTEGER */ CITY_KM_L;
attribute long /* INTEGER */ HIGHWAY_KM_L;
};

// ----- [STENE.GEAR]
interface STENE__GEAR(source relational wrDb2Volks
key (COD_E,DRIVE_TRAINS)
foreign_key(COD_E) references STENE__ENGINE
foreign_key(DRIVE_TRAINS,TRANS_GEAR_RATIO)
references STENE__DRIVE_TRAIN) {
// ResultSet [Index Info]

attribute char COD_E;
attribute char DRIVE_TRAINS;
attribute char TRANS_GEAR_RATIO;
attribute char FIRST;
attribute char SECOND;
attribute char THIRD;
attribute char FOURTH;
attribute char FIFTH;
attribute char REVERSE;
attribute char FINAL_DRIVE;
};

// ----- [STENE.MODEL]
interface STENE__MODEL(source relational wrDb2Volks
key (NAME,COD_E)
foreign_key(NAME) references STENE__CAR
foreign_key(COD_E) references STENE__ENGINE
foreign_key(STEERING) references STENE__STEERING) {
// ResultSet [Index Info]

attribute char NAME;

```

```

attribute char COD_E;
attribute char EPA_CLASS;
attribute long /* INTEGER */ PASSENGER_VOL_DM3;
attribute long /* INTEGER */ TRUNCK_VOL_DM3;
attribute long /* INTEGER */ SEATING_CAPACITY;
attribute long /* INTEGER */ VOL_FRONT_DM3;
attribute long /* INTEGER */ VOL_REAR_DM3;
attribute long /* DECIMAL */ HEAD_ROOM_FRONT_CM;
attribute long /* DECIMAL */ HEAD_ROOM_REAR_CM;
attribute long /* DECIMAL */ SHOULDER_ROOM_FRONT_CM;
attribute long /* DECIMAL */ SHOULDER_ROOM_REAR_CM;
attribute long /* DECIMAL */ LEG_ROOM_FRONT_CM;
attribute long /* DECIMAL */ LEG_ROOM_REAR_CM;
attribute char BODY_TYPE;
attribute char SUSPENSION_FRONT;
attribute char SUSPENSION_REAR;
attribute char SERVICE_BRAKE;
attribute char ANTI_LOCK_BRAK_SYS;
attribute char PARKING_BRAKE;
attribute char WHEELS;
attribute char TIRES;
attribute long /* DECIMAL */ DRAG_COEFFICIENT;
attribute char STEERING;
attribute long /* DECIMAL */ TURNS_LOCK2LOCK;
attribute long /* DECIMAL */ TURNING_CIRCLE_C2C_M;
attribute char RATIO;
attribute long /* DECIMAL */ PRICE;
};

// ----- [STENE.STEERING]
interface STENE__STEERING(source relational wrDb2Volks
key (STEERING)) {
// ResultSet [Index Info]

attribute char STEERING;
};

// ----- [STENE.WEIGHTS]
interface STENE__WEIGHTS(source relational wrDb2Volks
key (NAME,COD_E,DRIVE_TRAINS,TRANS_GEAR_RATIO)
foreign_key(CAR) references STENE__CAR

```

```

foreign_key(COD_E) references STENE___ENGINE
foreign_key(DRIVE_TRAINS,TRANS_GEAR_RATIO)
references STENE___DRIVE_TRAIN) {
// ResultSet [Index Info]

attribute char NAME;
attribute char COD_E;
attribute char DRIVE_TRAINS;
attribute char TRANS_GEAR_RATIO;
attribute long /* INTEGER */ CURB_KG;
attribute long /* INTEGER */ PAYLOAD_KG;
};

```

Renault

```

// Data source Description___ for []
// metadata
// ----- [STENE_CHANGE]
interface STENE_CHANGE(source relational wrDb2Renau
key (CODC,TYPE)) {
// ResultSet [Index Info]

attribute char CODC;
attribute char TYPE;
};

// ----- [STENE_MODEL]
interface STENE_MODEL(source relational wrDb2Renau
key (NAME)
foreign_key(NAMETSF) references STENE_TRANDSUS (NAMETS)
foreign_key(NAMETSR) references STENE_TRANDSUS (NAMETS)) {
// ResultSet [Index Info]

attribute char NAME;
attribute char NAMETSF;
attribute char NAMETSR;
attribute long /* INTEGER */ VOLBAG_DM3;
attribute long /* DECIMAL */ LENGTH;
attribute long /* DECIMAL */ WIDTH;
attribute long /* DECIMAL */ HEIGHT;
};

```



```

// ----- [STENE_MOTOR]
interface STENE_MOTOR(source relational wrDb2Renau
key (COD,TYPE_OF_INJECTION)) {
// ResultSet [Index Info]

attribute char COD;
attribute long /* INTEGER */ PISTON_DISPLACEMENT_CC;
attribute char BORE_X_RACE_MM;
attribute long /* INTEGER */ NO_OF_CYLINDER;
attribute long /* INTEGER */ NO_OF_VALVES;
attribute char COMPRESSION_RATIO;
attribute char POWER_KW;
attribute char POWER_CV;
attribute char BRACE_NM;
attribute char BRACE_MKG;
attribute char TYPE_OF_INJECTION;
attribute char FUEL;
};

// ----- [STENE_TRANSDUS]
interface STENE_TRANSDUS(source relational wrDb2Renau
key (NAMETS)) {
// ResultSet [Index Info]

attribute char NAMETS;
};

// ----- [STENE_VERSION]
interface STENE_VERSION(source relational wrDb2Renau
key (NAMEMODEL,NAMEVERSION)
foreign_key(COD,TYPE_OF_INJECTION) references STENE_MOTOR
foreign_key(NAMEMODEL) references STENE_MODEL (NAME)
foreign_key(CODC,TYPECHANGE) references STENE_CHANGE
(CODC,TYPE)) {
// ResultSet [Index Info]

attribute char COD;
attribute char NAMEMODEL;
attribute char TYPE_OF_INJECTION;
attribute char NAMEVERSION;
};

```

```
attribute char CODC;
attribute char TYPECHANGE;
attribute long /* INTEGER */ NO_OF_RELATIONSHIP;
attribute long /* DECIMAL */ VEL_I;
attribute long /* DECIMAL */ VEL_II;
attribute long /* DECIMAL */ VEL_III;
attribute long /* DECIMAL */ VEL_IV;
attribute long /* DECIMAL */ VEL_V;
attribute char WHEELS;
attribute char TIRES;
attribute long /* INTEGER */ ABILITY_TANK_L;
attribute char AERODYNAMICS;
attribute long /* DECIMAL */ DIA_OF_TURN_SIDEWALK;
attribute long /* DECIMAL */ DIA_OF_TURN_WALL;
attribute long /* DECIMAL */ NO_TURN_FLYING;
attribute long /* INTEGER */ SPEED_MAX;
attribute long /* DECIMAL */ P_0_100_S;
attribute long /* DECIMAL */ P_400M_S;
attribute long /* DECIMAL */ P_1000M_S;
attribute long /* DECIMAL */ CITY_CICLE;
attribute long /* DECIMAL */ CITY_EXTRA_CICLE;
attribute long /* DECIMAL */ MIXED_CICLE;
attribute long /* INTEGER */ CO2_G_KM;
attribute char REFRAINING_CIRCUIT;
attribute char ABS;
attribute char BRAKEBOOSTER;
attribute long /* INTEGER */ DIA_BRAKEBOOSTER;
attribute char FRONT;
attribute char REAR;
attribute long /* INTEGER */ W_EMPTY_MARCH_ORDER_KG;
attribute long /* INTEGER */ W_AUTHORIZED_MAX_KG;
attribute long /* INTEGER */ W_TRAVELLING_TOTAL_KG;
attribute long /* INTEGER */ W_PAYLOAD_KG;
attribute long /* INTEGER */ W_REFRAINED_TOWING_KG;
attribute long /* INTEGER */ W_NOT_REFRAINED_TOWING_KG;
attribute long /* DECIMAL */ PRICE;
};
```

Fiat

```
interface fiat
(source semistructured cardef_xml
extent fiat )
{
attribute set car;
};

interface car
(source semistructured cardef_xml
extent car )
{
attribute name name;
attribute engine engine;
attribute transmission transmission;
attribute steering steering;
attribute braking_system braking_system;
attribute suspension suspension;
attribute dimensions dimensions;
attribute string tires;
attribute elect_equipment elect_equipment;
attribute weights_capacities weights_capacities;
attribute performance performance;
attribute fuel_consumption fuel_consumption;
attribute string price;
};

interface engine
(source semistructured cardef_xml
extent engine )
{
attribute e_fuel e_fuel;
}
union engine1
{
attribute e_electrical e_electrical;
};

interface transmission
(source semistructured cardef_xml
extent transmission )
{
```

```
attribute string drive;
attribute speed speed;
};

interface steering
(source semistructured cardef_xml
extent steering )
{
attribute string s_type;
attribute string turning_circle;
};

interface braking_system
(source semistructured cardef_xml
extent braking_system )
{
attribute string b_front_mm;
attribute string b_rear_mm;
};

interface suspension
(source semistructured cardef_xml
extent suspension )
{
attribute string s_front;
attribute string s_rear;
};

interface dimensions
(source semistructured cardef_xml
extent dimensions )
{
attribute string length_m;
attribute string width_m;
attribute string height_m;
attribute string luggage_capacity_dm3;
attribute string wheelbase_m ?;
attribute string track_front_mm ?;
attribute string track_rear_mm ?;
};
```

```
interface elect_equipment
(source semistructured cardef_xml
extent elect_equipment )
{
attribute string battery_capacity;
attribute string alternator_current ?;
};
```

```
interface weights_capacities
(source semistructured cardef_xml
extent weights_capacities )
{
attribute string fuel_cap_l;
attribute string kerb_weight_kg;
attribute string towable_weight_kg ?;
};
```

```
interface performance
(source semistructured cardef_xml
extent performance )
{
attribute p_electrical p_electrical;
}
union performancel
{
attribute p_methane p_methane;
}
union performance2
{
attribute p_coupe p_coupe;
}
union performance3
{
attribute p_other p_other;
};
```

```
interface fuel_consumption
(source semistructured cardef_xml
extent fuel_consumption )
{
attribute string urban;
```

```
attribute string outoftown ?;
attribute string combined;
attribute string urban_range_km ?;
};

interface e_fuel
(source semistructured cardef_xml
extent e_fuel )
{
attribute string e_name;
attribute string cylinders ?;
attribute string layout ?;
attribute string bore_mm ?;
attribute string capacity_cc ?;
attribute string compression_ratio ?;
attribute string power_kw;
attribute string power_bhp;
attribute string torque_nm;
attribute string torque_kgm;
attribute string valve_gear ?;
attribute string fuel_system ?;
attribute string ignition ?;
};

interface e_electrical
(source semistructured cardef_xml
extent e_electrical )
{
attribute string e_name;
attribute string e_type;
attribute string layout;
attribute string power_kw;
attribute string torque_nm;
attribute string engine_control;
attribute string drive_battery_kwh;
attribute string bat_charge_time;
};

interface speed
(source semistructured cardef_xml
extent speed )
```

```
{
attribute string PCDATA_NODE;
}
union speed1
{
attribute gear gear;
};

interface p_electrical
(source semistructured cardef_xml
extent p_electrical )
{
attribute string top_speed;
attribute string p_0_50_km_h;
};

interface p_methane
(source semistructured cardef_xml
extent p_methane )
{
attribute string top_speed;
attribute string p_0_100_km_h;
attribute string p_0_1000m;
attribute string range_methane_km;
};

interface p_coupe
(source semistructured cardef_xml
extent p_coupe )
{
attribute string top_speed;
attribute string p_0_100_km_h;
attribute string p_0_1000m;
attribute string pickup_1000m_sec;
};

interface p_other
(source semistructured cardef_xml
extent p_other )
{
attribute string top_speed;
```

```

attribute string p_0_100_km_h ?;
attribute string p_0_1000m ?;
};

interface gear
(source semistructured cardef_xml
extent gear )
{
attribute string g1;
attribute string g2;
attribute string g3;
attribute string g4;
attribute string g5 ?;
attribute string g6 ?;
attribute string rm;
attribute string final_drive_ratio ?;
};

interface name
(source semistructured cardef_xml
extent name
key (name_id))
{
attribute string PCDATA_NODE;
attribute string name_id;
};

```

5.4 SI Designer

Una volta attivati i wrapper, e ottenuta la descrizione delle strutture delle sorgenti nel linguaggio del sistema si è passati alla fase di integrazione vera e propria con il sistema SI Designer: si sono caricati i wrapper come si vede in figura 5.3, quindi si è richiamato il modulo SIM, che come visto al capitolo tre estrae le relazioni intraschema, determinate dalle foreign key:

```

wrDb2Renau.STENE_MOTOR rt wrDb2Renau.STENE_VERSION;
wrDb2Renau.STENE_MODEL.NAME syn
wrDb2Renau.STENE_VERSION.NAMEMODEL;
wrDb2Renau.STENE_VERSION rt wrDb2Renau.STENE_MODEL;
wrDb2Renau.STENE_CHANGE rt wrDb2Renau.STENE_VERSION;

```

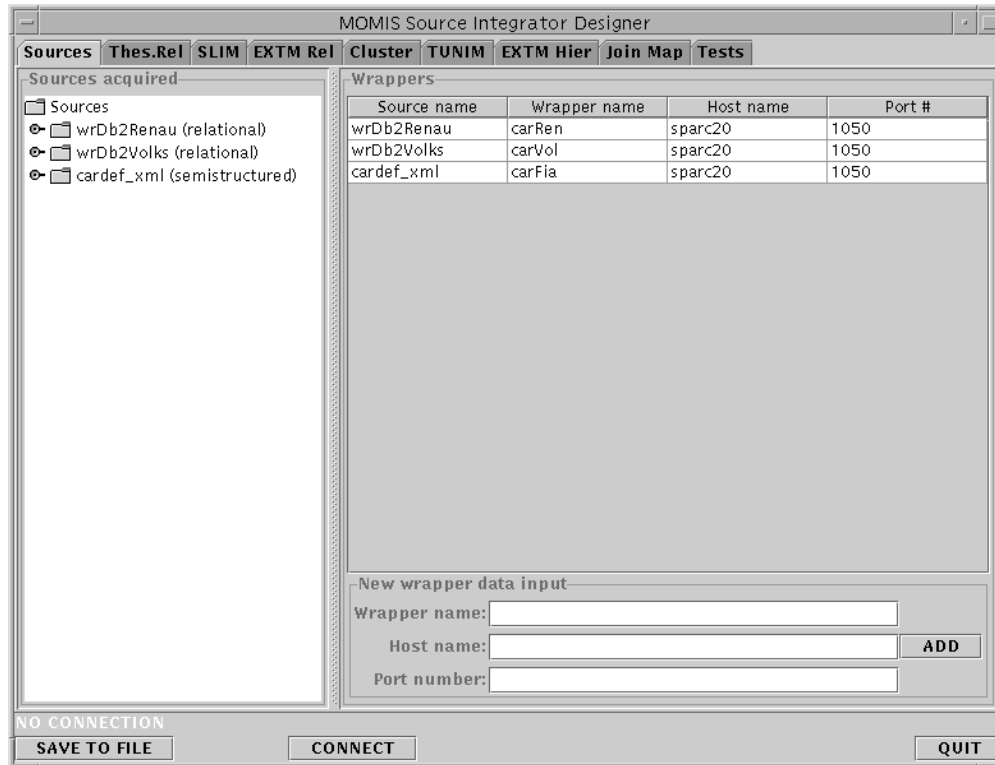



Figura 5.3: SI Designer: i wrapper

```

wrDb2Renau.STENE_TRANDSUS.NAMETS syn
wrDb2Renau.STENE_MODEL.NAMETSF;
wrDb2Renau.STENE_TRANDSUS rt wrDb2Renau.STENE_MODEL;
wrDb2Renau.STENE_TRANDSUS.NAMETS syn
wrDb2Renau.STENE_MODEL.NAMETSR;
wrDb2Volks.STENE__FUEL_CONSUMPTION rt
wrDb2Volks.STENE__CAR;
wrDb2Volks.STENE__FUEL_CONSUMPTION rt
wrDb2Volks.STENE__ENGINE;
wrDb2Volks.STENE__FUEL_CONSUMPTION rt
wrDb2Volks.STENE__DRIVE_TRAIN;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM rt
wrDb2Volks.STENE__ENGINE;
wrDb2Volks.STENE__MODEL rt wrDb2Volks.STENE__CAR;
wrDb2Volks.STENE__MODEL rt wrDb2Volks.STENE__ENGINE;
wrDb2Volks.STENE__STEERING rt wrDb2Volks.STENE__MODEL;
wrDb2Volks.STENE__WEIGHTS rt wrDb2Volks.STENE__CAR;

```

```

wrDb2Volks.STENE__WEIGHTS rt wrDb2Volks.STENE__ENGINE;
wrDb2Volks.STENE__WEIGHTS rt wrDb2Volks.STENE__DRIVE_TRAIN;
wrDb2Volks.STENE__GEAR rt wrDb2Volks.STENE__ENGINE;
wrDb2Volks.STENE__DRIVE_TRAIN rt wrDb2Volks.STENE__GEAR;
cardef_xml.car rt cardef_xml.fiat;
cardef_xml.braking_system rt cardef_xml.car;
cardef_xml.dimensions rt cardef_xml.car;
cardef_xml.elect_equipment rt cardef_xml.car;
cardef_xml.engine rt cardef_xml.car;
cardef_xml.fuel_consumption rt cardef_xml.car;
cardef_xml.name rt cardef_xml.car;
cardef_xml.performance rt cardef_xml.car;
cardef_xml.steering rt cardef_xml.car;
cardef_xml.suspension rt cardef_xml.car;
cardef_xml.transmission rt cardef_xml.car;
cardef_xml.weights_capacities rt cardef_xml.car;
cardef_xml.p_coupe rt cardef_xml.performance;
cardef_xml.p_electrical rt cardef_xml.performance;
cardef_xml.p_methane rt cardef_xml.performance;
cardef_xml.p_other rt cardef_xml.performance;
cardef_xml.gear rt cardef_xml.speed;
cardef_xml.e_electrical rt cardef_xml.engine;
cardef_xml.e_fuel rt cardef_xml.engine;
cardef_xml.speed rt cardef_xml.transmission;

```

Terminato questo passaggio si è passati al modulo SLIM.

5.4.1 L'uso di WordNet

L'uso del modulo SLIM è certamente, per il progettista, il passaggio più delicato del processo d'integrazione.

Nel nostro caso mi sono trovato di fronte al dover assegnare forme basi, quindi semplici, ai termini complessi che caratterizzano gli attributi delle sorgenti considerate.

Questo processo ha prodotto una semplificazione tale per cui svariati attributi si sono trovati ad avere una forma base simile, se non uguale, ad esempio:

```

codc, cod, cod_e = code;
type, e_type, s_type = type.

```

Questa uguaglianza, sommata a un uso diverso del medesimo termine negli schemi considerati, come:

renau.model che corrisponde alla classe volks.car, mentre la classe volks.model è la corrispondente della classe renau.version; ha portato alla generazione di oltre 1500 relazioni.

Da queste ho eliminato le relazioni di SYN costruite per errore, per i motivi appena considerati, e quelle di generalizzazione BT/NT costruite in virtù del fatto che la forma base di un attributo aveva un significato più generale di quello dell'altra coinvolta nella relazione, anche se ciò non corrispondeva alla realtà, come per esempio:

```
wrDb2Renau.STENE_MOTOR.NO_OF_VALVE nt
  cardef_xml.e_electrical.engine_control
```

che rappresentano uno il numero di valvole di un motore a carburante e l'altro il sistema di controllo di un motore elettrico, due attributi relativi al motore, ma assolutamente non in dipendenza l'uno dall'altro.

Al termine di questo processo di correzione sono rimaste le seguenti relazioni che sono poi state validate dal sistema e inserite a tutti gli effetti nel Thesaurus Comune, operazione necessaria per poterne sfruttare appieno la conoscenza per la formazione dei cluster e la fusione degli attributi negli schemi globali.

```
wrDb2Volks.STENE__ENGINE.KW rt
  wrDb2Volks.STENE__ENGINE.HORSEPOWER;
wrDb2Renau.STENE_VERSION.TIRES syn
  wrDb2Volks.STENE__MODEL.TIRES;
wrDb2Renau.STENE_VERSION.TIRES syn cardef_xml.car.tires;
wrDb2Volks.STENE__MODEL.TIRES syn cardef_xml.car.tires;
wrDb2Volks.STENE__FUEL_CONSUMPTION.DRIVE_TRAINS syn
  wrDb2Volks.STENE__DRIVE_TRAIN.DRIVE_TRAINS;
wrDb2Volks.STENE__FUEL_CONSUMPTION.DRIVE_TRAINS syn
  wrDb2Volks.STENE__WEIGHTS.DRIVE_TRAINS;
wrDb2Volks.STENE__FUEL_CONSUMPTION.DRIVE_TRAINS syn
  wrDb2Volks.STENE__GEAR.DRIVE_TRAINS;
wrDb2Volks.STENE__DRIVE_TRAIN.DRIVE_TRAINS syn
  wrDb2Volks.STENE__WEIGHTS.DRIVE_TRAINS;
wrDb2Volks.STENE__DRIVE_TRAIN.DRIVE_TRAINS syn
  wrDb2Volks.STENE__GEAR.DRIVE_TRAINS;
wrDb2Volks.STENE__WEIGHTS.DRIVE_TRAINS syn
  wrDb2Volks.STENE__GEAR.DRIVE_TRAINS;
wrDb2Volks.STENE__DRIVE_TRAIN rt
  wrDb2Volks.STENE__FUEL_CONSUMPTION.DRIVE_TRAINS;
wrDb2Volks.STENE__DRIVE_TRAIN rt
  wrDb2Volks.STENE__DRIVE_TRAIN.DRIVE_TRAINS;
```

```

wrDb2Volks.STENE___DRIVE_TRAIN rt
  wrDb2Volks.STENE___WEIGHTS.DRIVE_TRAINS;
wrDb2Volks.STENE___DRIVE_TRAIN rt
  wrDb2Volks.STENE___GEAR.DRIVE_TRAINS;
wrDb2Renau.STENE_VERSION.CITY_CICLE syn
  wrDb2Volks.STENE___FUEL_CONSUMPTION.CITY_KM_L;
wrDb2Renau.STENE_MODEL.LENGTH syn
  wrDb2Volks.STENE___CAR.LENGTH;
wrDb2Renau.STENE_MODEL.LENGTH syn
  cardef_xml.dimensions.length_m;
wrDb2Volks.STENE___CAR.LENGTH syn
  cardef_xml.dimensions.length_m;
cardef_xml.dimensions bt wrDb2Renau.STENE_MODEL.LENGTH;
wrDb2Renau.STENE_MODEL.LENGTH nt cardef_xml.car.dimensions;
cardef_xml.dimensions bt wrDb2Volks.STENE___CAR.LENGTH;
wrDb2Volks.STENE___CAR.LENGTH nt cardef_xml.car.dimensions;
cardef_xml.dimensions bt cardef_xml.dimensions.length_m;
cardef_xml.dimensions.length_m nt cardef_xml.car.dimensions;
wrDb2Renau.STENE_MODEL.LENGTH rt
  wrDb2Renau.STENE_MODEL.WIDTH;
wrDb2Renau.STENE_MODEL.LENGTH rt
  wrDb2Volks.STENE___CAR.WIDTH;
wrDb2Renau.STENE_MODEL.LENGTH rt
  cardef_xml.dimensions.width_m;
wrDb2Volks.STENE___CAR.LENGTH rt
  wrDb2Renau.STENE_MODEL.WIDTH;
wrDb2Volks.STENE___CAR.LENGTH rt
  wrDb2Volks.STENE___CAR.WIDTH;
wrDb2Volks.STENE___CAR.LENGTH rt
  cardef_xml.dimensions.width_m;
cardef_xml.dimensions.length_m rt
  wrDb2Renau.STENE_MODEL.WIDTH;
cardef_xml.dimensions.length_m rt
  wrDb2Volks.STENE___CAR.WIDTH;
cardef_xml.dimensions.length_m rt
  cardef_xml.dimensions.width_m;
wrDb2Renau.STENE_MODEL.LENGTH rt
  wrDb2Renau.STENE_MODEL.HEIGHT;
wrDb2Renau.STENE_MODEL.LENGTH rt
  wrDb2Volks.STENE___CAR.HEIGHT;
wrDb2Renau.STENE_MODEL.LENGTH rt

```

```

    cardef_xml.dimensions.height_m;
wrDb2Volks.STENE__CAR.LENGTH rt
    wrDb2Renau.STENE_MODEL.HEIGHT;
wrDb2Volks.STENE__CAR.LENGTH rt
    wrDb2Volks.STENE__CAR.HEIGHT;
wrDb2Volks.STENE__CAR.LENGTH rt
    cardef_xml.dimensions.height_m;
cardef_xml.dimensions.length_m rt
    wrDb2Renau.STENE_MODEL.HEIGHT;
cardef_xml.dimensions.length_m rt
    wrDb2Volks.STENE__CAR.HEIGHT;
cardef_xml.dimensions.length_m rt
    cardef_xml.dimensions.height_m;
cardef_xml.e_electrical syn
    cardef_xml.engine.e_electrical;
wrDb2Renau.STENE_MODEL.VOLBAG_DM3 syn
    cardef_xml.dimensions.luggage_capacity_dm3;
wrDb2Volks.STENE__FUEL_CONSUMPTION.TRANS_GEAR_RATIO syn
    wrDb2Volks.STENE__DRIVE_TRAIN.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__FUEL_CONSUMPTION.TRANS_GEAR_RATIO syn
    wrDb2Volks.STENE__WEIGHTS.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__FUEL_CONSUMPTION.TRANS_GEAR_RATIO syn
    wrDb2Volks.STENE__GEAR.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__DRIVE_TRAIN.TRANS_GEAR_RATIO syn
    wrDb2Volks.STENE__WEIGHTS.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__DRIVE_TRAIN.TRANS_GEAR_RATIO syn
    wrDb2Volks.STENE__GEAR.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__WEIGHTS.TRANS_GEAR_RATIO syn
    wrDb2Volks.STENE__GEAR.TRANS_GEAR_RATIO;
cardef_xml.transmission syn cardef_xml.car.transmission;
wrDb2Volks.STENE__GEAR bt
    wrDb2Volks.STENE__FUEL_CONSUMPTION.TRANS_GEAR_RATIO;
cardef_xml.gear bt
    wrDb2Volks.STENE__FUEL_CONSUMPTION.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__FUEL_CONSUMPTION.TRANS_GEAR_RATIO nt
    cardef_xml.speed.gear;
wrDb2Volks.STENE__GEAR bt
    wrDb2Volks.STENE__DRIVE_TRAIN.TRANS_GEAR_RATIO;
cardef_xml.gear bt
    wrDb2Volks.STENE__DRIVE_TRAIN.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__DRIVE_TRAIN.TRANS_GEAR_RATIO nt

```

```

    cardef_xml.speed.gear;
wrDb2Volks.STENE__GEAR bt
    wrDb2Volks.STENE__WEIGHTS.TRANS_GEAR_RATIO;
cardef_xml.gear bt
    wrDb2Volks.STENE__WEIGHTS.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__WEIGHTS.TRANS_GEAR_RATIO nt
    cardef_xml.speed.gear;
wrDb2Volks.STENE__GEAR bt
    wrDb2Volks.STENE__GEAR.TRANS_GEAR_RATIO;
cardef_xml.gear bt wrDb2Volks.STENE__GEAR.TRANS_GEAR_RATIO;
wrDb2Volks.STENE__GEAR.TRANS_GEAR_RATIO nt
    cardef_xml.speed.gear;
wrDb2Renau.STENE_MOTOR.NO_OF_VALVES syn
    wrDb2Volks.STENE__ENGINE.VALVE_TRAIN;
wrDb2Renau.STENE_MOTOR.NO_OF_VALVES syn
    cardef_xml.e_fuel.valve_gear;
wrDb2Volks.STENE__ENGINE.VALVE_TRAIN syn
    cardef_xml.e_fuel.valve_gear;
cardef_xml.fuel_consumption.urban_range_km rt
    cardef_xml.p_methane.range_methane_km;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM.ALTERNATOR_V_A syn
    wrDb2Volks.STENE__ENGINE.ALTERNATOR_V_A;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM.ALTERNATOR_V_A rt
    cardef_xml.elect_equipment.alternator_current;
wrDb2Volks.STENE__ENGINE.ALTERNATOR_V_A rt
    cardef_xml.elect_equipment.alternator_current;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM rt
    wrDb2Volks.STENE__ELECTRICAL_SYSTEM.ALTERNATOR_V_A;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM rt
    wrDb2Volks.STENE__ENGINE.ALTERNATOR_V_A;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM rt
    cardef_xml.elect_equipment.alternator_current;
wrDb2Renau.STENE_MOTOR.POWER_KW syn
    cardef_xml.e_electrical.power_kw;
wrDb2Renau.STENE_MOTOR.POWER_KW syn
    cardef_xml.e_fuel.power_kw;
wrDb2Renau.STENE_MOTOR.POWER_CV syn
    cardef_xml.e_fuel.power_bhp;
cardef_xml.e_electrical.power_kw syn
    cardef_xml.e_fuel.power_kw;
wrDb2Renau.STENE_MOTOR.POWER_KW rt

```

```

wrDb2Renau.STENE_MOTOR.POWER_CV;
wrDb2Renau.STENE_MOTOR.POWER_KW rt
  cardef_xml.e_fuel.power_bhp;
wrDb2Renau.STENE_MOTOR.POWER_CV rt
  cardef_xml.e_electrical.power_kw;
wrDb2Renau.STENE_MOTOR.POWER_CV rt
  cardef_xml.e_fuel.power_kw;
cardef_xml.e_electrical.power_kw rt
  cardef_xml.e_fuel.power_bhp;
wrDb2Renau.STENE_VERSION.TYPECHANGE syn
  wrDb2Renau.STENE_CHANGE.TYPE;
wrDb2Volks.STENE___ENGINE.TYPE syn
  cardef_xml.e_electrical.e_type;
wrDb2Renau.STENE_VERSION.FRONT syn
  wrDb2Volks.STENE___MODEL.SERVICE_BRAKE;
wrDb2Renau.STENE_VERSION.FRONT syn
  cardef_xml.braking_system.b_front_mm;
wrDb2Renau.STENE_VERSION.REAR syn
  wrDb2Volks.STENE___MODEL.SERVICE_BRAKE;
wrDb2Renau.STENE_VERSION.REAR syn
  cardef_xml.braking_system.b_rear_mm;
wrDb2Volks.STENE___MODEL.SERVICE_BRAKE syn
  cardef_xml.braking_system.b_rear_mm;
wrDb2Volks.STENE___MODEL.SERVICE_BRAKE syn
  cardef_xml.braking_system.b_front_mm;
wrDb2Renau.STENE_VERSION.FRONT rt
  wrDb2Renau.STENE_VERSION.REAR;
wrDb2Renau.STENE_VERSION.FRONT rt
  cardef_xml.braking_system.b_rear_mm;
wrDb2Renau.STENE_VERSION.REAR rt
  cardef_xml.braking_system.b_front_mm;
cardef_xml.braking_system.b_rear_mm rt
  cardef_xml.braking_system.b_front_mm;
wrDb2Renau.STENE_VERSION.VEL_III syn
  wrDb2Volks.STENE___GEAR.THIRD;
wrDb2Renau.STENE_VERSION.VEL_III syn cardef_xml.gear.g3;
wrDb2Volks.STENE___GEAR.THIRD syn cardef_xml.gear.g3;
wrDb2Volks.STENE___GEAR bt wrDb2Renau.STENE_VERSION.VEL_III;
cardef_xml.gear bt wrDb2Renau.STENE_VERSION.VEL_III;
wrDb2Renau.STENE_VERSION.VEL_III nt cardef_xml.speed.gear;
wrDb2Volks.STENE___GEAR bt wrDb2Volks.STENE___GEAR.THIRD;

```

```

cardef_xml.gear bt wrDb2Volks.STENE___GEAR.THIRD;
wrDb2Volks.STENE___GEAR.THIRD nt cardef_xml.speed.gear;
wrDb2Volks.STENE___GEAR bt cardef_xml.gear.g3;
cardef_xml.gear bt cardef_xml.gear.g3;
cardef_xml.gear.g3 nt cardef_xml.speed.gear;
wrDb2Renau.STENE_VERSION.VEL_III rt
  wrDb2Renau.STENE_VERSION.VEL_I;
wrDb2Renau.STENE_VERSION.VEL_III rt
  wrDb2Volks.STENE___GEAR.FIRST;
wrDb2Renau.STENE_VERSION.VEL_III rt cardef_xml.gear.g1;
wrDb2Volks.STENE___GEAR.THIRD rt
  wrDb2Renau.STENE_VERSION.VEL_I;
wrDb2Volks.STENE___GEAR.THIRD rt
  wrDb2Volks.STENE___GEAR.FIRST;
wrDb2Volks.STENE___GEAR.THIRD rt cardef_xml.gear.g1;
cardef_xml.gear.g3 rt wrDb2Renau.STENE_VERSION.VEL_I;
cardef_xml.gear.g3 rt wrDb2Volks.STENE___GEAR.FIRST;
cardef_xml.gear.g3 rt cardef_xml.gear.g1;
wrDb2Renau.STENE_VERSION.VEL_III rt
  wrDb2Volks.STENE___GEAR.REVERSE;
wrDb2Renau.STENE_VERSION.VEL_III rt cardef_xml.gear.rm;
wrDb2Volks.STENE___GEAR.THIRD rt
  wrDb2Volks.STENE___GEAR.REVERSE;
wrDb2Volks.STENE___GEAR.THIRD rt cardef_xml.gear.rm;
cardef_xml.gear.g3 rt wrDb2Volks.STENE___GEAR.REVERSE;
cardef_xml.gear.g3 rt cardef_xml.gear.rm;
wrDb2Renau.STENE_VERSION.VEL_III rt
  wrDb2Renau.STENE_VERSION.VEL_II;
wrDb2Renau.STENE_VERSION.VEL_III rt
  wrDb2Volks.STENE___GEAR.SECOND;
wrDb2Renau.STENE_VERSION.VEL_III rt cardef_xml.gear.g2;
wrDb2Volks.STENE___GEAR.THIRD rt
  wrDb2Renau.STENE_VERSION.VEL_II;
wrDb2Volks.STENE___GEAR.THIRD rt
  wrDb2Volks.STENE___GEAR.SECOND;
wrDb2Volks.STENE___GEAR.THIRD rt cardef_xml.gear.g2;
cardef_xml.gear.g3 rt wrDb2Renau.STENE_VERSION.VEL_II;
cardef_xml.gear.g3 rt wrDb2Volks.STENE___GEAR.SECOND;
cardef_xml.gear.g3 rt cardef_xml.gear.g2;
wrDb2Renau.STENE_VERSION.VEL_I syn
  wrDb2Volks.STENE___GEAR.FIRST;

```



```

wrDb2Renau.STENE_VERSION.VEL_I syn cardef_xml.gear.g1;
wrDb2Volks.STENE___GEAR.FIRST syn cardef_xml.gear.g1;
wrDb2Volks.STENE___GEAR bt wrDb2Renau.STENE_VERSION.VEL_I;
cardef_xml.gear bt wrDb2Renau.STENE_VERSION.VEL_I;
wrDb2Volks.STENE___GEAR bt wrDb2Volks.STENE___GEAR.FIRST;
cardef_xml.gear bt wrDb2Volks.STENE___GEAR.FIRST;
wrDb2Volks.STENE___GEAR.FIRST nt cardef_xml.speed.gear;
wrDb2Volks.STENE___GEAR bt cardef_xml.gear.g1;
cardef_xml.gear bt cardef_xml.gear.g1;
cardef_xml.gear.g1 nt cardef_xml.speed.gear;
wrDb2Renau.STENE_VERSION.VEL_I rt
    wrDb2Volks.STENE___GEAR.REVERSE;
wrDb2Renau.STENE_VERSION.VEL_I rt cardef_xml.gear.rm;
wrDb2Volks.STENE___GEAR.FIRST rt
    wrDb2Volks.STENE___GEAR.REVERSE;
wrDb2Volks.STENE___GEAR.FIRST rt cardef_xml.gear.rm;
cardef_xml.gear.g1 rt wrDb2Volks.STENE___GEAR.REVERSE;
cardef_xml.gear.g1 rt cardef_xml.gear.rm;
wrDb2Renau.STENE_VERSION.VEL_I rt
    wrDb2Renau.STENE_VERSION.VEL_II;
wrDb2Renau.STENE_VERSION.VEL_I rt
    wrDb2Volks.STENE___GEAR.SECOND;
wrDb2Renau.STENE_VERSION.VEL_I rt cardef_xml.gear.g2;
wrDb2Volks.STENE___GEAR.FIRST rt
    wrDb2Renau.STENE_VERSION.VEL_II;
wrDb2Volks.STENE___GEAR.FIRST rt
    wrDb2Volks.STENE___GEAR.SECOND;
wrDb2Volks.STENE___GEAR.FIRST rt cardef_xml.gear.g2;
cardef_xml.gear.g1 rt wrDb2Renau.STENE_VERSION.VEL_II;
cardef_xml.gear.g1 rt wrDb2Volks.STENE___GEAR.SECOND;
cardef_xml.gear.g1 rt cardef_xml.gear.g2;
cardef_xml.dimensions.track_front_mm rt
    cardef_xml.dimensions.track_rear_mm;
wrDb2Renau.STENE_VERSION.VEL_IV syn
    wrDb2Volks.STENE___GEAR.FOURTH;
wrDb2Renau.STENE_VERSION.VEL_IV syn cardef_xml.gear.g4;
wrDb2Volks.STENE___GEAR.FOURTH syn cardef_xml.gear.g4;
wrDb2Renau.STENE_VERSION.VEL_IV rt
    wrDb2Renau.STENE_VERSION.VEL_V;
wrDb2Renau.STENE_VERSION.VEL_IV rt
    wrDb2Volks.STENE___GEAR.FIFTH;

```

```
wrDb2Renau.STENE_VERSION.VEL_IV rt cardef_xml.gear.g5;
wrDb2Volks.STENE___GEAR.FOURTH rt
    wrDb2Renau.STENE_VERSION.VEL_V;
wrDb2Volks.STENE___GEAR.FOURTH rt
    wrDb2Volks.STENE___GEAR.FIFTH;
wrDb2Volks.STENE___GEAR.FOURTH rt cardef_xml.gear.g5;
cardef_xml.gear.g4 rt wrDb2Renau.STENE_VERSION.VEL_V;
cardef_xml.gear.g4 rt wrDb2Volks.STENE___GEAR.FIFTH;
cardef_xml.gear.g4 rt cardef_xml.gear.g5;
wrDb2Renau.STENE_VERSION.VEL_IV rt cardef_xml.gear.g6;
wrDb2Volks.STENE___GEAR.FOURTH rt cardef_xml.gear.g6;
cardef_xml.gear.g4 rt cardef_xml.gear.g6;
cardef_xml.fuel_consumption syn
    cardef_xml.car.fuel_consumption;
cardef_xml.braking_system syn cardef_xml.car.braking_system;
cardef_xml.car.braking_system rt
    wrDb2Volks.STENE___MODEL.PARKING_BRAKE;
cardef_xml.braking_system rt
    wrDb2Volks.STENE___MODEL.PARKING_BRAKE;
cardef_xml.e_fuel.fuel_system rt
    wrDb2Volks.STENE___ENGINE.COOLING_SYSTEM;
cardef_xml.e_fuel.fuel_system rt
    wrDb2Volks.STENE___ENGINE.COOLING_SYSTEM_L;
wrDb2Volks.STENE___MODEL.LEG_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.HEAD_ROOM_REAR_CM;
wrDb2Volks.STENE___MODEL.LEG_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.LEG_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.LEG_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.HEAD_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.LEG_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.LEG_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_REAR_CM;
wrDb2Volks.STENE___MODEL.HEAD_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.LEG_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.HEAD_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.HEAD_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.HEAD_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.HEAD_ROOM_REAR_CM rt
    wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_REAR_CM;
```

```

wrDb2Volks.STENE___MODEL.LEG_ROOM_FRONT_CM rt
wrDb2Volks.STENE___MODEL.HEAD_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.LEG_ROOM_FRONT_CM rt
wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.LEG_ROOM_FRONT_CM rt
wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_REAR_CM;
wrDb2Volks.STENE___MODEL.HEAD_ROOM_FRONT_CM rt
wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_FRONT_CM;
wrDb2Volks.STENE___MODEL.HEAD_ROOM_FRONT_CM rt
wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_REAR_CM;
wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_FRONT_CM rt
wrDb2Volks.STENE___MODEL.SHOULDER_ROOM_REAR_CM;
wrDb2Renau.STENE_VERSION.ABILITY_TANK_L syn
wrDb2Volks.STENE___ENGINE.FUEL_TANK_L;
wrDb2Renau.STENE_VERSION.SPEED_MAX syn
cardef_xml.p_coupe.top_speed;
wrDb2Renau.STENE_VERSION.SPEED_MAX syn
cardef_xml.p_electrical.top_speed;
wrDb2Renau.STENE_VERSION.SPEED_MAX syn
cardef_xml.p_methane.top_speed;
wrDb2Renau.STENE_VERSION.SPEED_MAX syn
cardef_xml.p_other.top_speed;
cardef_xml.p_coupe.top_speed syn
cardef_xml.p_electrical.top_speed;
cardef_xml.p_coupe.top_speed syn
cardef_xml.p_methane.top_speed;
cardef_xml.p_coupe.top_speed syn
cardef_xml.p_other.top_speed;
cardef_xml.p_electrical.top_speed syn
cardef_xml.p_methane.top_speed;
cardef_xml.p_electrical.top_speed syn
cardef_xml.p_other.top_speed;
cardef_xml.p_methane.top_speed syn
cardef_xml.p_other.top_speed;
cardef_xml.e_electrical.layout syn
cardef_xml.e_fuel.layout;
wrDb2Volks.STENE___ELECTRICAL_SYSTEM.BATTERY_A_HRS syn
wrDb2Volks.STENE___ENGINE.BATTERY_A_HRS;
wrDb2Volks.STENE___ELECTRICAL_SYSTEM.BATTERY_A_HRS syn
cardef_xml.elect_equipment.battery_capacity;
wrDb2Volks.STENE___ELECTRICAL_SYSTEM.BATTERY_V syn

```

```

wrDb2Volks.STENE__ENGINE.BATTERY_V;
wrDb2Volks.STENE__ENGINE.BATTERY_A_HRS syn
  cardef_xml.elect_equipment.battery_capacity;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM.BATTERY_A_HRS rt
  wrDb2Volks.STENE__ELECTRICAL_SYSTEM.BATTERY_V;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM.BATTERY_A_HRS rt
  wrDb2Volks.STENE__ENGINE.BATTERY_V;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM.BATTERY_V rt
  wrDb2Volks.STENE__ENGINE.BATTERY_A_HRS;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM.BATTERY_V rt
  cardef_xml.elect_equipment.battery_capacity;
wrDb2Volks.STENE__ENGINE.BATTERY_A_HRS rt
  wrDb2Volks.STENE__ENGINE.BATTERY_V;
wrDb2Volks.STENE__ENGINE.BATTERY_V rt
  cardef_xml.elect_equipment.battery_capacity;
wrDb2Renau.STENE_VERSION.PRICE syn
  wrDb2Volks.STENE__MODEL.PRICE;
wrDb2Renau.STENE_VERSION.PRICE syn cardef_xml.car.price;
wrDb2Volks.STENE__MODEL.PRICE syn cardef_xml.car.price;
cardef_xml.engine syn cardef_xml.car.engine;
wrDb2Volks.STENE__GEAR.REVERSE syn cardef_xml.gear.rm;
wrDb2Volks.STENE__GEAR bt wrDb2Volks.STENE__GEAR.REVERSE;
cardef_xml.gear bt wrDb2Volks.STENE__GEAR.REVERSE;
wrDb2Volks.STENE__GEAR.REVERSE nt cardef_xml.speed.gear;
wrDb2Volks.STENE__GEAR bt cardef_xml.gear.rm;
cardef_xml.gear bt cardef_xml.gear.rm;
cardef_xml.gear.rm nt cardef_xml.speed.gear;
wrDb2Volks.STENE__GEAR.REVERSE rt
  wrDb2Renau.STENE_VERSION.VEL_II;
wrDb2Volks.STENE__GEAR.REVERSE rt
  wrDb2Volks.STENE__GEAR.SECOND;
wrDb2Volks.STENE__GEAR.REVERSE rt cardef_xml.gear.g2;
cardef_xml.gear.rm rt wrDb2Renau.STENE_VERSION.VEL_II;
cardef_xml.gear.rm rt wrDb2Volks.STENE__GEAR.SECOND;
cardef_xml.gear.rm rt cardef_xml.gear.g2;
wrDb2Renau.STENE_VERSION.NAMEMODEL syn
  wrDb2Volks.STENE__CAR.NAME;
wrDb2Renau.STENE_VERSION.NAMEMODEL syn
  wrDb2Volks.STENE__FUEL_CONSUMPTION.NAME;
wrDb2Renau.STENE_VERSION.NAMEMODEL syn
  wrDb2Volks.STENE__MODEL.NAME;

```

```

wrDb2Renau.STENE_VERSION.NAMEMODEL syn
    vwrDb2Volks.STENE__WEIGHTS.NAME;
wrDb2Renau.STENE_VERSION.NAMEMODEL syn cardef_xml.car.name;
wrDb2Renau.STENE_VERSION.NAMEMODEL syn
    cardef_xml.name.PCDATA_NODE;
wrDb2Renau.STENE_MODEL.NAME syn wrDb2Volks.STENE__CAR.NAME;
wrDb2Renau.STENE_MODEL.NAME syn
    wrDb2Volks.STENE__FUEL_CONSUMPTION.NAME;
wrDb2Renau.STENE_MODEL.NAME syn
    wrDb2Volks.STENE__MODEL.NAME;
wrDb2Renau.STENE_MODEL.NAME syn
    wrDb2Volks.STENE__WEIGHTS.NAME;
wrDb2Renau.STENE_MODEL.NAME syn
    cardef_xml.name.PCDATA_NODE;
wrDb2Volks.STENE__CAR.NAME syn
    wrDb2Volks.STENE__FUEL_CONSUMPTION.NAME;
wrDb2Volks.STENE__CAR.NAME syn
    wrDb2Volks.STENE__MODEL.NAME;
wrDb2Volks.STENE__CAR.NAME syn
    wrDb2Volks.STENE__WEIGHTS.NAME;
wrDb2Volks.STENE__CAR.NAME syn
    cardef_xml.name.PCDATA_NODE;
wrDb2Volks.STENE__FUEL_CONSUMPTION.NAME syn
    wrDb2Volks.STENE__MODEL.NAME;
wrDb2Volks.STENE__FUEL_CONSUMPTION.NAME syn
    wrDb2Volks.STENE__WEIGHTS.NAME;
wrDb2Volks.STENE__FUEL_CONSUMPTION.NAME syn
    cardef_xml.name.PCDATA_NODE;
wrDb2Volks.STENE__MODEL.NAME syn
    wrDb2Volks.STENE__WEIGHTS.NAME;
wrDb2Volks.STENE__MODEL.NAME syn
    cardef_xml.name.PCDATA_NODE;
wrDb2Volks.STENE__WEIGHTS.NAME syn
    cardef_xml.name.PCDATA_NODE;
cardef_xml.name syn cardef_xml.car.name;
cardef_xml.e_electrical.e_name syn cardef_xml.e_fuel.e_name;
wrDb2Renau.STENE_VERSION.NO_OF_RELATIONSHIP syn
    cardef_xml.speed.PCDATA_NODE;
wrDb2Volks.STENE__GEAR syn cardef_xml.gear;
wrDb2Volks.STENE__GEAR syn cardef_xml.speed.gear;
cardef_xml.gear syn cardef_xml.speed.gear;

```

```

wrDb2Volks.STENE__GEAR rt
  wrDb2Renau.STENE_VERSION.NO_OF_RELATIONSHIP;
cardef_xml.gear rt
  wrDb2Renau.STENE_VERSION.NO_OF_RELATIONSHIP;
wrDb2Renau.STENE_VERSION.NO_OF_RELATIONSHIP rt
  cardef_xml.speed.gear;
wrDb2Volks.STENE__GEAR rt cardef_xml.speed.PCDATA_NODE;
cardef_xml.gear rt cardef_xml.speed.PCDATA_NODE;
cardef_xml.speed.gear rt cardef_xml.speed.PCDATA_NODE;
cardef_xml.gear.g6 rt wrDb2Renau.STENE_VERSION.VEL_V;
cardef_xml.gear.g6 rt wrDb2Volks.STENE__GEAR.FIFTH;
cardef_xml.gear.g6 rt cardef_xml.gear.g5;
wrDb2Renau.STENE_MODEL.HEIGHT syn
  wrDb2Volks.STENE__CAR.HEIGHT;
wrDb2Renau.STENE_MODEL.HEIGHT syn
  cardef_xml.dimensions.height_m;
wrDb2Volks.STENE__CAR.HEIGHT syn
  cardef_xml.dimensions.height_m;
cardef_xml.dimensions bt wrDb2Renau.STENE_MODEL.HEIGHT;
wrDb2Renau.STENE_MODEL.HEIGHT nt cardef_xml.car.dimensions;
cardef_xml.dimensions bt wrDb2Volks.STENE__CAR.HEIGHT;
wrDb2Volks.STENE__CAR.HEIGHT nt cardef_xml.car.dimensions;
cardef_xml.dimensions bt cardef_xml.dimensions.height_m;
cardef_xml.dimensions.height_m nt cardef_xml.car.dimensions;
wrDb2Renau.STENE_MODEL.HEIGHT rt
  wrDb2Renau.STENE_MODEL.WIDTH;
wrDb2Renau.STENE_MODEL.HEIGHT rt
  wrDb2Volks.STENE__CAR.WIDTH;
wrDb2Renau.STENE_MODEL.HEIGHT rt
  cardef_xml.dimensions.width_m;
wrDb2Volks.STENE__CAR.HEIGHT rt
  wrDb2Renau.STENE_MODEL.WIDTH;
wrDb2Volks.STENE__CAR.HEIGHT rt
  wrDb2Volks.STENE__CAR.WIDTH;
wrDb2Volks.STENE__CAR.HEIGHT rt
  cardef_xml.dimensions.width_m;
cardef_xml.dimensions.height_m rt
  wrDb2Renau.STENE_MODEL.WIDTH;
cardef_xml.dimensions.height_m rt
  wrDb2Volks.STENE__CAR.WIDTH;
cardef_xml.dimensions.height_m rt

```

```

cardef_xml.dimensions.width_m;
wrDb2Renau.STENE_VERSION.TYPE_OF_INJECTION syn
wrDb2Volks.STENE___ELECTRICAL_SYSTEM.IGNITION;
wrDb2Renau.STENE_VERSION.TYPE_OF_INJECTION syn
wrDb2Volks.STENE___ENGINE.IGNITION;
wrDb2Renau.STENE_VERSION.TYPE_OF_INJECTION syn
cardef_xml.e_fuel.ignition;
wrDb2Volks.STENE___ELECTRICAL_SYSTEM.IGNITION syn
wrDb2Volks.STENE___ENGINE.IGNITION;
wrDb2Volks.STENE___ELECTRICAL_SYSTEM.IGNITION syn
cardef_xml.e_fuel.ignition;
wrDb2Volks.STENE___ENGINE.IGNITION syn
cardef_xml.e_fuel.ignition;
wrDb2Renau.STENE_MOTOR.FUEL syn
wrDb2Volks.STENE___ENGINE.FUEL_REQUIREMENT;
cardef_xml.e_fuel syn cardef_xml.engine.e_fuel;
wrDb2Renau.STENE_MOTOR.FUEL rt cardef_xml.engine.e_fuel;
cardef_xml.e_fuel rt wrDb2Renau.STENE_MOTOR.FUEL;
wrDb2Volks.STENE___ENGINE.FUEL_REQUIREMENT rt
cardef_xml.engine.e_fuel;
cardef_xml.e_fuel rt
wrDb2Volks.STENE___ENGINE.FUEL_REQUIREMENT;
wrDb2Volks.STENE___CAR.FRONT_TRACK rt
wrDb2Volks.STENE___CAR.REAR_TRACK;
wrDb2Volks.STENE___MODEL.VOL_REAR_DM3 rt
wrDb2Volks.STENE___MODEL.VOL_FRONT_DM3;
wrDb2Volks.STENE___MODEL.VOL_REAR_DM3 rt
wrDb2Volks.STENE___MODEL.PASSENGER_VOL_DM3;
wrDb2Volks.STENE___MODEL.VOL_FRONT_DM3 rt
wrDb2Volks.STENE___MODEL.PASSENGER_VOL_DM3;
wrDb2Renau.STENE_MOTOR.BRACE_MKG rt w
wrDb2Renau.STENE_MOTOR.BRACE_NM;
wrDb2Renau.STENE_MOTOR.NO_OF_CYLINDER syn
cardef_xml.e_fuel.cylinders;
wrDb2Renau.STENE_MOTOR.NO_OF_CYLINDER rt
wrDb2Volks.STENE___ENGINE.CYLINDER_BLOCK;
cardef_xml.e_fuel.cylinders rt
wrDb2Volks.STENE___ENGINE.CYLINDER_BLOCK;
wrDb2Renau.STENE_MOTOR.COMPRESSION_RATIO syn
wrDb2Volks.STENE___ENGINE.COMPRESSION_RATIO;
wrDb2Renau.STENE_MOTOR.COMPRESSION_RATIO syn

```

```

    cardef_xml.e_fuel.compression_ratio;
wrDb2Volks.STENE__ENGINE.COMPRESSION_RATIO syn
    cardef_xml.e_fuel.compression_ratio;
wrDb2Volks.STENE__ENGINE.COOLING_SYSTEM rt
    wrDb2Volks.STENE__ENGINE.COOLING_SYSTEM_L;
wrDb2Volks.STENE__CAR.WHEELBASE syn
    cardef_xml.dimensions.wheelbase_m;
wrDb2Renau.STENE_TRANDSUS syn cardef_xml.car.suspension;
wrDb2Renau.STENE_TRANDSUS syn cardef_xml.suspension;
wrDb2Renau.STENE_TRANDSUS.NAMETS syn
    wrDb2Volks.STENE__MODEL.SUSPENSION_REAR;
wrDb2Renau.STENE_TRANDSUS.NAMETS syn
    wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT;
wrDb2Renau.STENE_TRANDSUS.NAMETS syn
    cardef_xml.suspension.s_front;
wrDb2Renau.STENE_TRANDSUS.NAMETS syn
    cardef_xml.suspension.s_rear;
wrDb2Renau.STENE_MODEL.NAMETSR syn
    wrDb2Volks.STENE__MODEL.SUSPENSION_REAR;
wrDb2Renau.STENE_MODEL.NAMETSR syn
    cardef_xml.suspension.s_rear;
wrDb2Renau.STENE_MODEL.NAMETSF syn
    wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT;
wrDb2Renau.STENE_MODEL.NAMETSF syn
    cardef_xml.suspension.s_front;
wrDb2Volks.STENE__MODEL.SUSPENSION_REAR syn
    cardef_xml.suspension.s_rear;
wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT syn
    cardef_xml.suspension.s_front;
cardef_xml.suspension syn cardef_xml.car.suspension;
wrDb2Renau.STENE_TRANDSUS rt
    wrDb2Renau.STENE_TRANDSUS.NAMETS;
wrDb2Renau.STENE_TRANDSUS rt
    wrDb2Renau.STENE_MODEL.NAMETSR;
wrDb2Renau.STENE_TRANDSUS rt
    wrDb2Renau.STENE_MODEL.NAMETSF;
wrDb2Renau.STENE_TRANDSUS rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_REAR;
wrDb2Renau.STENE_TRANDSUS rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT;
wrDb2Renau.STENE_TRANDSUS rt

```



```

    cardef_xml.suspension.s_front;
wrDb2Renau.STENE_TRANDSUS rt
    cardef_xml.suspension.s_rear;
wrDb2Renau.STENE_TRANDSUS.NAMETS rt
    cardef_xml.car.suspension;
cardef_xml.suspension rt wrDb2Renau.STENE_TRANDSUS.NAMETS;
wrDb2Renau.STENE_MODEL.NAMETSR rt
    wrDb2Renau.STENE_MODEL.NAMETSF;
wrDb2Renau.STENE_MODEL.NAMETSR rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT;
wrDb2Renau.STENE_MODEL.NAMETSR rt cardef_xml.car.suspension;
cardef_xml.suspension rt wrDb2Renau.STENE_MODEL.NAMETSR;
wrDb2Renau.STENE_MODEL.NAMETSR rt
    cardef_xml.suspension.s_front;
wrDb2Renau.STENE_MODEL.NAMETSF rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_REAR;
wrDb2Renau.STENE_MODEL.NAMETSF rt cardef_xml.car.suspension;
cardef_xml.suspension rt wrDb2Renau.STENE_MODEL.NAMETSF;
wrDb2Renau.STENE_MODEL.NAMETSF rt
    cardef_xml.suspension.s_rear;
wrDb2Volks.STENE__MODEL.SUSPENSION_REAR rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT;
wrDb2Volks.STENE__MODEL.SUSPENSION_REAR rt
    cardef_xml.car.suspension;
cardef_xml.suspension rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_REAR;
wrDb2Volks.STENE__MODEL.SUSPENSION_REAR rt
    cardef_xml.suspension.s_front;
wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT rt
    cardef_xml.car.suspension;
cardef_xml.suspension rt
    wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT;
wrDb2Volks.STENE__MODEL.SUSPENSION_FRONT rt
    cardef_xml.suspension.s_rear;
cardef_xml.car.suspension rt cardef_xml.suspension.s_front;
cardef_xml.car.suspension rt cardef_xml.suspension.s_rear;
cardef_xml.suspension rt cardef_xml.suspension.s_front;
cardef_xml.suspension rt cardef_xml.suspension.s_rear;
cardef_xml.suspension.s_front rt
    cardef_xml.suspension.s_rear;
wrDb2Renau.STENE_MOTOR.COD syn

```

```

wrDb2Renau.STENE_VERSION.COD;
wrDb2Renau.STENE_VERSION.CODC syn
wrDb2Renau.STENE_CHANGE.CODC;
wrDb2Volks.STENE___FUEL_CONSUMPTION.COD_E syn
wrDb2Volks.STENE___ENGINE.COD_E;
wrDb2Volks.STENE___FUEL_CONSUMPTION.COD_E syn
wrDb2Volks.STENE___MODEL.COD_E;
wrDb2Volks.STENE___FUEL_CONSUMPTION.COD_E syn
wrDb2Volks.STENE___WEIGHTS.COD_E;
wrDb2Volks.STENE___FUEL_CONSUMPTION.COD_E syn
wrDb2Volks.STENE___GEAR.COD_E;
wrDb2Volks.STENE___ENGINE.COD_E syn
wrDb2Volks.STENE___MODEL.COD_E;
wrDb2Volks.STENE___ENGINE.COD_E syn
wrDb2Volks.STENE___WEIGHTS.COD_E;
wrDb2Volks.STENE___ENGINE.COD_E syn
wrDb2Volks.STENE___GEAR.COD_E;
wrDb2Volks.STENE___MODEL.COD_E syn
wrDb2Volks.STENE___WEIGHTS.COD_E;
wrDb2Volks.STENE___MODEL.COD_E syn
wrDb2Volks.STENE___GEAR.COD_E;
wrDb2Volks.STENE___WEIGHTS.COD_E syn
wrDb2Volks.STENE___GEAR.COD_E;
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG syn
cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Volks.STENE___WEIGHTS syn
cardef_xml.car.weights_capacities;
wrDb2Volks.STENE___WEIGHTS syn
cardef_xml.weights_capacities;
cardef_xml.weights_capacities syn
cardef_xml.car.weights_capacities;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;

```

```
wrDb2Volks.STENE___WEIGHTS rt
  wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
  cardef_xml.car.weights_capacities;
cardef_xml.weights_capacities rt
  wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG;
wrDb2Renau.STENE_VERSION.W_AUTHORIZED_MAX_KG rt
  cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG rt
  wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG;
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG rt
  wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG rt
  wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG;
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG rt
  wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;
wrDb2Volks.STENE___WEIGHTS rt
  wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG;
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG rt
  cardef_xml.car.weights_capacities;
cardef_xml.weights_capacities rt
  wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG;
wrDb2Renau.STENE_VERSION.W_PAYLOAD_KG rt
  cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG rt
  wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG rt
  wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG;
wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG rt
  wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;
wrDb2Volks.STENE___WEIGHTS rt
  wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG;
wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG rt
  cardef_xml.car.weights_capacities;
cardef_xml.weights_capacities rt
  wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG;
wrDb2Renau.STENE_VERSION.W_EMPTY_MARCH_ORDER_KG rt
  cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG rt
  wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG;
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG rt
```

```

wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;
wrDb2Volks.STENE___WEIGHTS rt
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG rt
cardef_xml.car.weights_capacities;
cardef_xml.weights_capacities rt
wrDb2Renau.STENE_VERSION.W_NOT_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG rt
wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;
wrDb2Volks.STENE___WEIGHTS rt
wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG;
wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG rt
cardef_xml.car.weights_capacities;
cardef_xml.weights_capacities rt
wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG;
wrDb2Renau.STENE_VERSION.W_TRAVELLING_TOTAL_KG rt
cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Volks.STENE___WEIGHTS rt
wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG rt
cardef_xml.car.weights_capacities;
cardef_xml.weights_capacities rt
wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG;
wrDb2Renau.STENE_VERSION.W_REFRAINED_TOWING_KG rt
cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Volks.STENE___WEIGHTS rt
cardef_xml.weights_capacities.towable_weight_kg;
cardef_xml.car.weights_capacities rt
cardef_xml.weights_capacities.towable_weight_kg;
cardef_xml.weights_capacities.towable_weight_kg;
cardef_xml.weights_capacities rt
cardef_xml.weights_capacities.towable_weight_kg;
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_MKG syn
cardef_xml.e_fuel.torque_kgm;
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_NM syn
cardef_xml.e_electrical.torque_nm;
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_NM syn
cardef_xml.e_fuel.torque_nm;
cardef_xml.e_electrical.torque_nm syn
cardef_xml.e_fuel.torque_nm;
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_MKG rt
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_NM;

```

```

wrDb2Volks.STENE___ENGINE.MAX_TORQUE_MKG rt
  cardef_xml.e_electrical.torque_nm;
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_MKG rt
  cardef_xml.e_fuel.torque_nm;
wrDb2Volks.STENE___ENGINE.MAX_TORQUE_NM rt
  cardef_xml.e_fuel.torque_kgm;
cardef_xml.e_electrical.torque_nm rt
  cardef_xml.e_fuel.torque_kgm;
cardef_xml.e_fuel.torque_kgm rt cardef_xml.e_fuel.torque_nm;
cardef_xml.car syn cardef_xml.fiat.car;
wrDb2Renau.STENE_VERSION.P_1000M_S syn
  cardef_xml.p_coupe.p_0_1000m;
wrDb2Renau.STENE_VERSION.P_1000M_S syn
  cardef_xml.p_methane.p_0_1000m;
wrDb2Renau.STENE_VERSION.P_1000M_S syn
  cardef_xml.p_other.p_0_1000m;
wrDb2Renau.STENE_VERSION.P_0_100_S syn
  cardef_xml.p_coupe.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_0_100_S syn
  cardef_xml.p_methane.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_0_100_S syn
  cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_coupe syn cardef_xml.performance.p_coupe;
cardef_xml.p_coupe syn cardef_xml.performance.p_electrical;
cardef_xml.p_coupe syn cardef_xml.performance.p_methane;
cardef_xml.p_coupe syn cardef_xml.performance.p_other;
cardef_xml.p_coupe syn cardef_xml.p_electrical;
cardef_xml.p_coupe syn cardef_xml.p_other;
cardef_xml.p_coupe.p_0_100_km_h syn
  cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_coupe.p_0_100_km_h syn
  cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_coupe.p_0_1000m syn
  cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_coupe.p_0_1000m syn
  cardef_xml.p_other.p_0_1000m;
cardef_xml.performance syn cardef_xml.car.performance;
cardef_xml.performance.p_coupe syn
  cardef_xml.performance.p_electrical;
cardef_xml.performance.p_coupe syn
  cardef_xml.performance.p_methane;

```

```

cardef_xml.performance.p_coupe syn
  cardef_xml.performance.p_other;
cardef_xml.p_electrical syn cardef_xml.performance.p_coupe;
cardef_xml.p_other syn cardef_xml.performance.p_coupe;
cardef_xml.performance.p_electrical syn
  cardef_xml.performance.p_methane;
cardef_xml.performance.p_electrical syn
  cardef_xml.performance.p_other;
cardef_xml.p_electrical syn
  cardef_xml.performance.p_electrical;
cardef_xml.p_other syn cardef_xml.performance.p_electrical;
cardef_xml.performance.p_methane syn
  cardef_xml.performance.p_other;
cardef_xml.p_electrical syn cardef_xml.performance.p_methane;
cardef_xml.p_other syn cardef_xml.performance.p_methane;
cardef_xml.p_electrical syn cardef_xml.performance.p_other;
cardef_xml.p_other syn cardef_xml.performance.p_other;
cardef_xml.p_electrical syn cardef_xml.p_other;
cardef_xml.p_methane.p_0_100_km_h syn
  cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_methane.p_0_1000m syn
  cardef_xml.p_other.p_0_1000m;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  wrDb2Renau.STENE_VERSION.P_0_100_S;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  wrDb2Renau.STENE_VERSION.P_400M_S;
cardef_xml.p_coupe rt wrDb2Renau.STENE_VERSION.P_1000M_S;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.p_coupe.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.p_coupe.pickup_1000m_sec;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.car.performance;
cardef_xml.performance rt wrDb2Renau.STENE_VERSION.P_1000M_S;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.performance.p_coupe;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.performance.p_electrical;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.performance.p_methane;
wrDb2Renau.STENE_VERSION.P_1000M_S rt

```

```
cardef_xml.performance.p_other;
cardef_xml.p_electrical rt
  wrDb2Renau.STENE_VERSION.P_1000M_S;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.p_electrical.p_0_50_km_h;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_other rt wrDb2Renau.STENE_VERSION.P_1000M_S;
wrDb2Renau.STENE_VERSION.P_1000M_S rt
  cardef_xml.p_other.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  wrDb2Renau.STENE_VERSION.P_400M_S;
cardef_xml.p_coupe rt wrDb2Renau.STENE_VERSION.P_0_100_S;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.p_coupe.pickup_1000m_sec;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.p_coupe.p_0_1000m;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.car.performance;
cardef_xml.performance rt wrDb2Renau.STENE_VERSION.P_0_100_S;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.performance.p_coupe;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.performance.p_electrical;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.performance.p_methane;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.performance.p_other;
cardef_xml.p_electrical rt
  wrDb2Renau.STENE_VERSION.P_0_100_S;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.p_electrical.p_0_50_km_h;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt wrDb2Renau.STENE_VERSION.P_0_100_S;
wrDb2Renau.STENE_VERSION.P_0_100_S rt
  cardef_xml.p_other.p_0_1000m;
cardef_xml.p_coupe rt wrDb2Renau.STENE_VERSION.P_400M_S;
wrDb2Renau.STENE_VERSION.P_400M_S rt
  cardef_xml.p_coupe.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_400M_S rt
```

```

    cardef_xml.p_coupe.pickup_1000m_sec;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.p_coupe.p_0_1000m;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.car.performance;
cardef_xml.performance rt
    wrDb2Renau.STENE_VERSION.P_400M_S;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.performance.p_coupe;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.performance.p_electrical;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.performance.p_methane;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.performance.p_other;
cardef_xml.p_electrical rt
    wrDb2Renau.STENE_VERSION.P_400M_S;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.p_electrical.p_0_50_km_h;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.p_methane.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt wrDb2Renau.STENE_VERSION.P_400M_S;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.p_other.p_0_100_km_h;
wrDb2Renau.STENE_VERSION.P_400M_S rt
    cardef_xml.p_other.p_0_1000m;
cardef_xml.p_coupe rt cardef_xml.p_coupe.p_0_100_km_h;
cardef_xml.p_coupe rt cardef_xml.p_coupe.pickup_1000m_sec;
cardef_xml.p_coupe rt cardef_xml.p_coupe.p_0_1000m;
cardef_xml.p_coupe rt cardef_xml.car.performance;
cardef_xml.p_coupe rt cardef_xml.performance;
cardef_xml.p_coupe rt cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.p_coupe rt cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_coupe rt cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_coupe rt cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_coupe rt cardef_xml.p_other.p_0_1000m;
cardef_xml.p_coupe.p_0_100_km_h rt
    cardef_xml.p_coupe.pickup_1000m_sec;
cardef_xml.p_coupe.p_0_100_km_h rt

```



```
cardef_xml.p_coupe.p_0_1000m;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.car.performance;
cardef_xml.performance rt cardef_xml.p_coupe.p_0_100_km_h;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.performance.p_coupe;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.performance.p_electrical;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.performance.p_methane;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.performance.p_other;
cardef_xml.p_electrical rt cardef_xml.p_coupe.p_0_100_km_h;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt cardef_xml.p_coupe.p_0_100_km_h;
cardef_xml.p_coupe.p_0_100_km_h rt
  cardef_xml.p_other.p_0_1000m;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.p_coupe.p_0_1000m;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.car.performance;
cardef_xml.performance rt
  cardef_xml.p_coupe.pickup_1000m_sec;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.performance.p_coupe;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.performance.p_electrical;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.performance.p_methane;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.performance.p_other;
cardef_xml.p_electrical rt
  cardef_xml.p_coupe.pickup_1000m_sec;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_coupe.pickup_1000m_sec rt
```

```
cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt cardef_xml.p_coupe.pickup_1000m_sec;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.p_other.p_0_100_kmh;
cardef_xml.p_coupe.pickup_1000m_sec rt
  cardef_xml.p_other.p_0_1000m;
cardef_xml.p_coupe.p_0_1000m rt cardef_xml.car.performance;
cardef_xml.performance rt cardef_xml.p_coupe.p_0_1000m;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.performance.p_coupe;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.performance.p_electrical;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.performance.p_methane;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.performance.p_other;
cardef_xml.p_electrical rt cardef_xml.p_coupe.p_0_1000m;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.p_electrical.p_0_50_kmh;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.p_methane.p_0_100_kmh;
cardef_xml.p_other rt cardef_xml.p_coupe.p_0_1000m;
cardef_xml.p_coupe.p_0_1000m rt
  cardef_xml.p_other.p_0_100_kmh;
cardef_xml.car.performance rt
  cardef_xml.performance.p_coupe;
cardef_xml.car.performance rt
  cardef_xml.performance.p_electrical;
cardef_xml.car.performance rt
  cardef_xml.performance.p_methane;
cardef_xml.car.performance rt
  cardef_xml.performance.p_other;
cardef_xml.p_electrical rt cardef_xml.car.performance;
cardef_xml.car.performance rt
  cardef_xml.p_electrical.p_0_50_kmh;
cardef_xml.car.performance rt
  cardef_xml.p_methane.p_0_100_kmh;
cardef_xml.car.performance rt
  cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt cardef_xml.car.performance;
cardef_xml.car.performance rt
```

```
cardef_xml.p_other.p_0_100_km_h;
cardef_xml.car.performance rt
  cardef_xml.p_other.p_0_1000m;
cardef_xml.performance rt cardef_xml.performance.p_coupe;
cardef_xml.performance rt
  cardef_xml.performance.p_electrical;
cardef_xml.performance rt cardef_xml.performance.p_methane;
cardef_xml.performance rt cardef_xml.performance.p_other;
cardef_xml.performance rt cardef_xml.p_electrical;
cardef_xml.performance rt
  cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.performance rt
  cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.performance rt cardef_xml.p_methane.p_0_1000m;
cardef_xml.performance rt cardef_xml.p_other;
cardef_xml.performance rt cardef_xml.p_other.p_0_100_km_h;
cardef_xml.performance rt cardef_xml.p_other.p_0_1000m;
cardef_xml.performance.p_coupe rt
  cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.performance.p_coupe rt
  cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.performance.p_coupe rt
  cardef_xml.p_methane.p_0_1000m;
cardef_xml.performance.p_coupe rt
  cardef_xml.p_other.p_0_100_km_h;
cardef_xml.performance.p_coupe rt
  cardef_xml.p_other.p_0_1000m;
cardef_xml.performance.p_electrical rt
  cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.performance.p_electrical rt
  cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.performance.p_electrical rt
  cardef_xml.p_methane.p_0_1000m;
cardef_xml.performance.p_electrical rt
  cardef_xml.p_other.p_0_100_km_h;
cardef_xml.performance.p_electrical rt
  cardef_xml.p_other.p_0_1000m;
cardef_xml.performance.p_methane rt
  cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.performance.p_methane rt
  cardef_xml.p_methane.p_0_100_km_h;
```

cardef_xml.performance.p_methane rt
cardef_xml.p_methane.p_0_1000m;
cardef_xml.performance.p_methane rt
cardef_xml.p_other.p_0_100_km_h;
cardef_xml.performance.p_methane rt
cardef_xml.p_other.p_0_1000m;
cardef_xml.performance.p_other rt
cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.performance.p_other rt
cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.performance.p_other rt
cardef_xml.p_methane.p_0_1000m;
cardef_xml.performance.p_other rt
cardef_xml.p_other.p_0_100_km_h;
cardef_xml.performance.p_other rt
cardef_xml.p_other.p_0_1000m;
cardef_xml.p_electrical rt
cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.p_electrical rt
cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_electrical rt cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_electrical rt cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_electrical rt cardef_xml.p_other.p_0_1000m;
cardef_xml.p_electrical.p_0_50_km_h rt
cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_electrical.p_0_50_km_h rt
cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt cardef_xml.p_electrical.p_0_50_km_h;
cardef_xml.p_electrical.p_0_50_km_h rt
cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_electrical.p_0_50_km_h rt
cardef_xml.p_other.p_0_1000m;
cardef_xml.p_methane.p_0_100_km_h rt
cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_other rt cardef_xml.p_methane.p_0_100_km_h;
cardef_xml.p_methane.p_0_100_km_h rt
cardef_xml.p_other.p_0_1000m;
cardef_xml.p_other rt cardef_xml.p_methane.p_0_1000m;
cardef_xml.p_methane.p_0_1000m rt
cardef_xml.p_other.p_0_100_km_h;
cardef_xml.p_other rt cardef_xml.p_other.p_0_100_km_h;

```

cardef_xml.p_other rt cardef_xml.p_other.p_0_1000m;
cardef_xml.p_other.p_0_100_km_h rt
  cardef_xml.p_other.p_0_1000m;
wrDb2Renau.STENE_VERSION.WHEELS syn
  wrDb2Volks.STENE___MODEL.WHEELS;
wrDb2Renau.STENE_VERSION.VEL_II syn
  wrDb2Volks.STENE___GEAR.SECOND;
wrDb2Renau.STENE_VERSION.VEL_II syn cardef_xml.gear.g2;
wrDb2Volks.STENE___GEAR.SECOND syn cardef_xml.gear.g2;
wrDb2Volks.STENE___GEAR bt wrDb2Renau.STENE_VERSION.VEL_II;
cardef_xml.gear bt wrDb2Renau.STENE_VERSION.VEL_II;
wrDb2Renau.STENE_VERSION.VEL_II nt cardef_xml.speed.gear;
wrDb2Volks.STENE___GEAR bt wrDb2Volks.STENE___GEAR.SECOND;
cardef_xml.gear bt wrDb2Volks.STENE___GEAR.SECOND;
wrDb2Volks.STENE___GEAR.SECOND nt cardef_xml.speed.gear;
wrDb2Volks.STENE___GEAR bt cardef_xml.gear.g2;
cardef_xml.gear bt cardef_xml.gear.g2;
cardef_xml.gear.g2 nt cardef_xml.speed.gear;
wrDb2Renau.STENE_MODEL.WIDTH syn
  wrDb2Volks.STENE___CAR.WIDTH;
wrDb2Renau.STENE_MODEL.WIDTH syn
  cardef_xml.dimensions.width_m;
wrDb2Volks.STENE___CAR.WIDTH syn
  cardef_xml.dimensions.width_m;
cardef_xml.dimensions bt wrDb2Renau.STENE_MODEL.WIDTH;
wrDb2Renau.STENE_MODEL.WIDTH nt cardef_xml.car.dimensions;
cardef_xml.dimensions bt wrDb2Volks.STENE___CAR.WIDTH;
wrDb2Volks.STENE___CAR.WIDTH nt cardef_xml.car.dimensions;
cardef_xml.dimensions bt cardef_xml.dimensions.width_m;
cardef_xml.dimensions.width_m nt cardef_xml.car.dimensions;
cardef_xml.dimensions syn cardef_xml.car.dimensions;
wrDb2Renau.STENE_VERSION.VEL_V syn
  wrDb2Volks.STENE___GEAR.FIFTH;
wrDb2Renau.STENE_VERSION.VEL_V syn cardef_xml.gear.g5;
wrDb2Volks.STENE___GEAR.FIFTH syn cardef_xml.gear.g5;
wrDb2Renau.STENE_MOTOR.BORE_X_RACE_MM syn
  wrDb2Volks.STENE___ENGINE.BORE_MM;
wrDb2Renau.STENE_MOTOR.BORE_X_RACE_MM syn
  cardef_xml.e_fuel.bore_mm;
wrDb2Volks.STENE___ENGINE.BORE_MM syn
  cardef_xml.e_fuel.bore_mm;

```

5.4.2 L'arricchimento del progettista

Per migliorare la formazione dei cluster e per ottimizzare la fusione mi sono trovato in alcuni casi a dover tornare al Thes.Rel per aggiungere relazioni che non erano state trovate nelle fasi precedenti. Queste sono le relazioni che ho introdotto io:

```
wrDb2Renau.STENE_VERSION.VEL_I nt cardef_xml.speed.gear;
wrDb2Volks.STENE__STEERING syn cardef_xml.steering;
wrDb2Volks.STENE__STEERING.STEERING syn
    cardef_xml.steering.s_type;
wrDb2Volks.STENE__ENGINE syn wrDb2Renau.STENE_MOTOR;
wrDb2Volks.STENE__ENGINE syn cardef_xml.e_fuel;
wrDb2Volks.STENE__ENGINE syn cardef_xml.e_electrical;
wrDb2Renau.STENE_MOTOR syn cardef_xml.e_electrical;
wrDb2Renau.STENE_MOTOR syn cardef_xml.e_fuel;
cardef_xml.p_methane syn cardef_xml.p_other;
cardef_xml.p_methane syn cardef_xml.p_electrical;
cardef_xml.p_methane syn cardef_xml.p_coupe;
wrDb2Renau.STENE_VERSION syn wrDb2Volks.STENE__MODEL;
wrDb2Renau.STENE_MODEL syn wrDb2Volks.STENE__CAR;
wrDb2Volks.STENE__ELECTRICAL_SYSTEM syn
    cardef_xml.elect_equipment;
wrDb2Volks.STENE__FUEL_CONSUMPTION syn
    cardef_xml.fuel_consumption;
cardef_xml.name nt cardef_xml.car;
cardef_xml.dimensions nt wrDb2Volks.STENE__CAR;
cardef_xml.dimensions nt wrDb2Renau.STENE_MODEL;
cardef_xml.gear nt wrDb2Renau.STENE_VERSION;
wrDb2Volks.STENE__GEAR nt wrDb2Renau.STENE_VERSION;
wrDb2Volks.STENE__FUEL_CONSUMPTION nt
    wrDb2Renau.STENE_VERSION;
cardef_xml.fuel_consumption nt wrDb2Renau.STENE_VERSION;
cardef_xml.braking_system nt wrDb2Renau.STENE_VERSION;
cardef_xml.braking_system nt wrDb2Volks.STENE__MODEL;
cardef_xml.speed nt wrDb2Renau.STENE_VERSION;
wrDb2Volks.STENE__DRIVE_TRAIN nt
    wrDb2Renau.STENE_VERSION;
cardef_xml.suspension nt wrDb2Volks.STENE__MODEL;
cardef_xml.transmission.drive syn
    wrDb2Volks.STENE__DRIVE_TRAIN.DRIVE_TRAINS;
cardef_xml.transmission nt
    wrDb2Volks.STENE__DRIVE_TRAIN;
```

```
cardef_xml.weights_capacities nt
  wrDb2Renau.STENE_VERSION;
wrDb2Volks.STENE___WEIGHTS nt wrDb2Renau.STENE_VERSION;
wrDb2Renau.STENE_MOTOR.COD syn
  wrDb2Volks.STENE___ENGINE.COD_E;
wrDb2Renau.STENE_MOTOR.PISTON_DISPLACEMENT_CC syn
  wrDb2Volks.STENE___ENGINE.DISPLACEMENT_CC;
wrDb2Renau.STENE_MOTOR.POWER_KW syn
  wrDb2Volks.STENE___ENGINE.KW;
wrDb2Renau.STENE_MOTOR.POWER_CV syn
  wrDb2Volks.STENE___ENGINE.HORSEPOWER;
wrDb2Renau.STENE_MOTOR.BRACE_NM syn
  wrDb2Volks.STENE___ENGINE.MAX_TORQUE_NM;
wrDb2Renau.STENE_MOTOR.BRACE_MKG syn
  wrDb2Volks.STENE___ENGINE.MAX_TORQUE_MKG;
wrDb2Renau.STENE_MOTOR.TYPE_OF_INJECTION syn
  wrDb2Volks.STENE___ENGINE.IGNITION;
cardef_xml.e_fuel.fuel_system syn
  wrDb2Volks.STENE___ENGINE.FUEL_REQUIREMENT;
cardef_xml.e_fuel.power_bhp syn
  wrDb2Volks.STENE___ENGINE.HORSEPOWER;
cardef_xml.e_fuel.power_kw syn
  wrDb2Volks.STENE___ENGINE.KW;
cardef_xml.e_fuel.capacity_cc syn
  wrDb2Volks.STENE___ENGINE.DISPLACEMENT_CC;
cardef_xml.e_fuel.capacity_cc syn
  wrDb2Renau.STENE_MOTOR.PISTON_DISPLACEMENT_CC;
cardef_xml.e_fuel.e_name syn wrDb2Renau.STENE_MOTOR.COD;
cardef_xml.e_fuel.torque_nm syn
  wrDb2Renau.STENE_MOTOR.BRACE_NM;
cardef_xml.e_fuel.torque_kgm syn
  wrDb2Renau.STENE_MOTOR.BRACE_MKG;
cardef_xml.e_fuel.ignition syn
  wrDb2Renau.STENE_MOTOR.TYPE_OF_INJECTION;
cardef_xml.e_fuel.fuel_system syn
  wrDb2Renau.STENE_MOTOR.FUEL;
cardef_xml.e_electrical.e_name syn
  wrDb2Renau.STENE_MOTOR.COD;
cardef_xml.e_electrical.torque_nm syn
  wrDb2Renau.STENE_MOTOR.BRACE_NM;
cardef_xml.suspension nt wrDb2Renau.STENE_MODEL;
```

```

cardef_xml.name nt wrDb2Renau.STENE_MODEL;
cardef_xml.name nt wrDb2Volks.STENE__CAR;
wrDb2Volks.STENE__MODEL.ANTI_LOCK_BRAK_SYS syn
    wrDb2Renau.STENE_VERSION.ABS;
wrDb2Renau.STENE_VERSION.ABILITY_TANK_L syn
    cardef_xml.weights_capacities.fuel_cap_l;
wrDb2Volks.STENE__ENGINE.FUEL_TANK_L syn
    cardef_xml.weights_capacities.fuel_cap_l;
wrDb2Volks.STENE__MODEL.SERVICE_BRAKE syn
    wrDb2Renau.STENE_VERSION.BRAKEBOOSTER;
cardef_xml.fuel_consumption.outoftown syn
    wrDb2Renau.STENE_VERSION.CITY_EXTRA_CICLE;
cardef_xml.fuel_consumption.urban syn
    wrDb2Renau.STENE_VERSION.CITY_CICLE;
cardef_xml.fuel_consumption.combined syn
    wrDb2Renau.STENE_VERSION.MIXED_CICLE;
wrDb2Volks.STENE__FUEL_CONSUMPTION.HIGHWAY_KM_L syn
    wrDb2Renau.STENE_VERSION.CITY_EXTRA_CICLE;
wrDb2Volks.STENE__FUEL_CONSUMPTION.HIGHWAY_KM_L syn
    cardef_xml.fuel_consumption.outoftown;
wrDb2Volks.STENE__FUEL_CONSUMPTION.CITY_KM_L syn
    cardef_xml.fuel_consumption.urban.

```

Nonostante abbia dovuto inserire queste relazioni manualmente, risulta evidente che le aggiunte che deve fare il progettista sono poche se confrontate con tutte le relazioni estratte automaticamente dal sistema.

5.4.3 I cluster

Il passaggio successivo nel processo di integrazione è, come abbiamo visto al capitolo tre, la formazione dei cluster da parte del modulo ARTEMIS, che sfruttando le affinità tra gli attributi e le classi raggruppa le classi locali in cluster. Nel nostro caso ho utilizzato i coefficienti di affinità standard visti al paragrafo 3.4, così facendo il sistema ha generato una buona organizzazione di cluster, nella quale, però, la classe `cardef_xml.bracking_system` veniva mappata in un cluster dedicato, mentre i suoi attributi sono sinonimi di attributi della tabella Version del database Renault; inoltre abbiamo che `cardef_xml.bracking_system` è NT di `wrDb2Renau.STENE_VERSION`.

Per rimediare a questo problema ci sono due possibilità:

1. intervenire manualmente rimuovendo il cluster e mappando la classe `cardef_xml.bracking_system` nel cluster corretto;

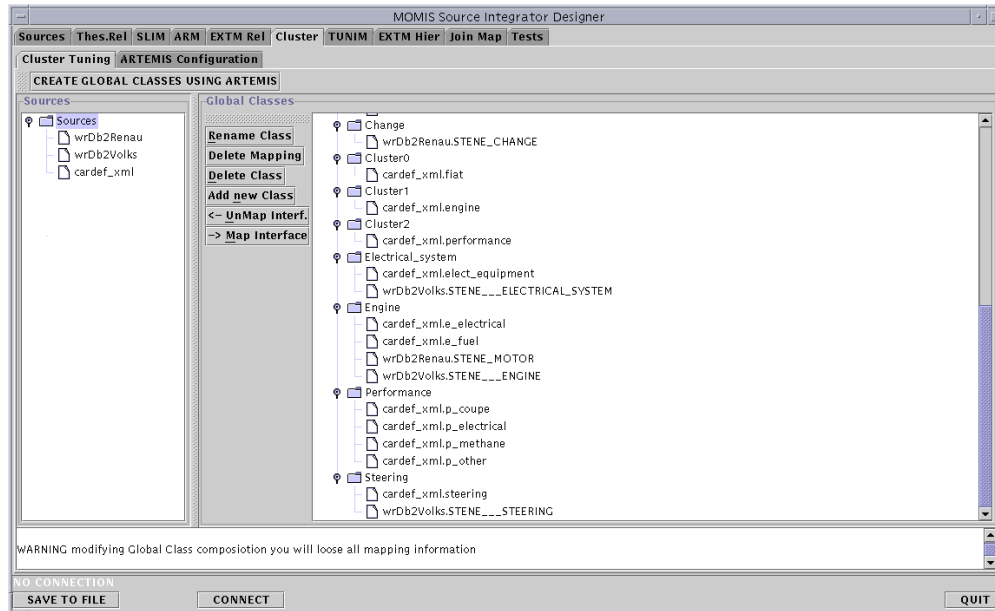


Figura 5.4: SI Designer: ARTEMIS

2. modificare i coefficienti di ARTEMIS e fargli ripetere il procedimento.

Col secondo metodo per ottenere il risultato voluto ho dovuto alzare a 0,9 il peso delle relazioni NT.

Bisogna sottolineare alcune particolarità dei cluster generati:

- la maggior parte delle interfacce XML che non contengono dati, ma riferimenti a sottostrutture contenenti dati sono state isolate con un cluster per ognuna;
- in virtù del fatto che le strutture dati erano state costruite in maniera diversa, e gli attributi assegnati in modo non corrispondente tra le classi locali dei diversi database, si è generato un cluster di grosse dimensioni contenente svariate classi locali.

Questi nel dettaglio i cluster generati:

```
+ Car
cardef_xml.braking_system
cardef_xml.car
cardef_xml.dimension
cardef_xml.fuel_consumption
cardef_xml.gear
```

```
cardef_xml.name
cardef_xml.speed
cardef_xml.suspension
cardef_xml.transmission
cardef_xml.weights_capacities
wrDb2Renau.STENE_MODEL
wrDb2Renau.STENE_TRANDSUS
wrDb2Renau.STENE_VERSION
wrDb2Volks.STENE___CAR
wrDb2Volks.STENE___DRIVE_TRAIN
wrDb2Volks.STENE___FUEL_CONSUMPTION
wrDb2Volks.STENE___GEAR
wrDb2Volks.STENE___MODEL
wrDb2Volks.STENE___WEIGHTS

+ Change
wrDb2Renau.STENE_CHANGE

+ Cluster0
cardef_xml.fiat

+ Cluster1
cardef_xml.engine

+ Cluster2
cardef_xml.performance

+ Electrical_system
cardef_xml.e_electrical
cardef_xml.e_fuel
wrDb2Renau.STENE_MOTOR
wrDb2Volks.STENE___ENGINE

+ Performance
cardef_xml.p_coupe
cardef_xml.p_electrical
cardef_xml.p_methane
cardef_xml.p_other

+ Steering
cardef_xml.steering
```

```
wrDb2Volks.STENE___STEERING
```

5.4.4 Le mapping table

L'ultimo passo del processo di integrazione consiste nella generazione dell mapping table e nella fusione degli attributi.

Dopo aver rinominato i cluster più significativi sono passato alle mapping table, generate automaticamente in funzione dei cluster, e qui ho compiuto la fusione degli attributi.

Per completare correttamente tutte le fusioni possibili in modo automatico, sono tornato indietro, alcune volte, al Thes.Rel per inserire alcune delle relazioni citate al paragrafo 5.4.2 e che come detto non erano state inserite dal sistema. Il processo ha portato in quasi tutti i casi all'eliminazione degli attributi globali ridondanti, nella classe globale *Car* non sono invece riuscito a raggiungere l'ottimo per un problema già noto dell'algoritmo di fusione.

Come detto al paragrafo precedente l'integrazione ha portato alla formazione di una classe globale, *Car* in cui sono state raccolte molte classi ciò ha prodotto un'enorme tabella in cui la maggior parte degli elementi hanno valor NULL; per quanto poco gradevole da vedere questa tabella riporta comunque un risultato corretto. Questa situazione che si è creata è dovuta al ricchissimo insieme di attributi con cui sono state descritte le automobili della Volkswagen rispetto a quelle della Renault e della Fiat, e alla compattazione di molti attributi delle auto della Renault nella classe locale *Version*.

Queste le classi globali, rappresentate in linguaggio ODL_{T3} :

```
interface Car
{
  attribute string AERODYNAMICS;
  mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANSDUS: NULL
  )
}
```

```

        wrDb2Renau.STENE_VERSION: AERODYNAMICS
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string ANTI_LOCK_BRAK_SYS;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: ABS
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: ANTI_LOCK_BRAK_SYS
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string BODY_TYPE;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL

```

```

    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: BODY_TYPE
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string BRAKEBOOSTER;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: BRAKEBOOSTER
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer CO2_G_KM;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL

```

```

        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: CO2_G_KM
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string COD;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: COD
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute string CODC;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL

```

```

    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: CODC
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string COD_E3;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: COD_E
        wrDb2Volks.STENE__GEAR: COD_E
        wrDb2Volks.STENE__MODEL: COD_E
        wrDb2Volks.STENE__WEIGHTS: COD_E
    )
attribute integer CURB_KG;
mapping_rules(

```

```

        cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: CURB_KG
    )
attribute integer DIA_BRAKEBOOSTER;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: DIA_BRAKEBOOSTER
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL

```



```

    )
attribute integer DIA_OF_TURN_SIDEWALK;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: DIA_OF_TURN_SIDEWALK
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer DIA_OF_TURN_WALL;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: DIA_OF_TURN_WALL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL

```

```

        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer DRAG_COEFFICIENT;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: DRAG_COEFFICIENT
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute string EPA_CLASS;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL

```

```

        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: EPA_CLASS
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string FINAL_DRIVE;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: FINAL_DRIVE
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string FRONT;
mapping_rules(
    cardef_xml.braking_system: b_rear_mm
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL

```

```

        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: SERVICE_BRAKE
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer FRONT_TRACK;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions:
        cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: FRONT
        wrDb2Volks.STENE__CAR: FRONT_TRACK
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer GROUND_CLEARANCE;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL

```

```

        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: GROUND_CLEARANCE
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer HEAD_ROOM_FRONT_CM;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: HEAD_ROOM_FRONT_CM
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer HEAD_ROOM_REAR_CM;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL

```

```

        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
        wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: HEAD_ROOM_REAR_CM
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer LEG_ROOM_FRONT_CM;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: LEG_ROOM_FRONT_CM
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute integer LEG_ROOM_REAR_CM;
mapping_rules(
    cardef_xml.braking_system: NULL

```

```

        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: LEG_ROOM_REAR_CM
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer LENGTH;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: LENGTH
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: LENGTH
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )

```

```
attribute string NAMETSF;
  mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: s_front
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NAMETSF
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
  )
attribute string NAMETSR1;
  mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NAMETS
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
```



```

        wrDb2Volks.STENE___MODEL: SUSPENSION_REAR
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string NAMETSR2;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NAMETSR
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute string NAMEVERSION;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NAMEVERSION
    wrDb2Volks.STENE___CAR: NULL

```

```

        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer NO_TURN_FLYING;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANSUS: NULL
        wrDb2Renau.STENE_VERSION: NO_TURN_FLYING
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string PARKING_BRAKE;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL

```

```

        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: PARKING_BRAKE
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer PASSENGER_VOL_DM3;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: PASSENGER_VOL_DM3
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer PAYLOAD_KG;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL

```

```

        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: PAYLOAD_KG
    )
attribute string PCDATA_NODE1;
    mapping_rules(
        cardef_xml.braking_system: NULL
            cardef_xml.car: NULL
            cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: PCDATA_NODE
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
            wrDb2Renau.STENE_MODEL: NULL
            wrDb2Renau.STENE_TRANDSUS: NULL
            wrDb2Renau.STENE_VERSION: NO_OF_RELATIONSHIP
            wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer P_0_100_S;
    mapping_rules(
        cardef_xml.braking_system: NULL
            cardef_xml.car: NULL
            cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL

```

```

        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: P_0_100_S
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer P_1000M_S;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: P_1000M_S
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute integer P_400M_S;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL

```

```

        cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: NULL
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
            wrDb2Renau.STENE_MODEL: NULL
            wrDb2Renau.STENE_TRANDSUS: NULL
            wrDb2Renau.STENE_VERSION: P_400M_S
            wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string RATIO;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
            cardef_xml.dimensions: NULL
            cardef_xml.fuel_consumption: NULL
                cardef_xml.gear: NULL
                cardef_xml.name: NULL
                cardef_xml.speed: NULL
                cardef_xml.suspension: NULL
                cardef_xml.transmission: NULL
            cardef_xml.weights_capacities: NULL
                wrDb2Renau.STENE_MODEL: NULL
                wrDb2Renau.STENE_TRANDSUS: NULL
                wrDb2Renau.STENE_VERSION: NULL
                wrDb2Volks.STENE___CAR: NULL
            wrDb2Volks.STENE___DRIVE_TRAIN: NULL
        wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
            wrDb2Volks.STENE___GEAR: NULL
            wrDb2Volks.STENE___MODEL: RATIO
            wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string REAR;

```



```
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string REFRAINING_CIRCUIT;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: REFRAINING_CIRCUIT
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute string REVERSE;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
```



```

wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: REVERSE
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute integer SEATING_CAPACITY;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: SEATING_CAPACITY
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute string SECOND;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL

```

```

        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: SECOND
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer SHOULDER_ROOM_FRONT_CM;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: SHOULDER_ROOM_FRONT_CM
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer SHOULDER_ROOM_REAR_CM;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL

```

```

    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: SHOULDER_ROOM_REAR_CM
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer SPEED_MAX;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: SPEED_MAX
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string STEERING;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL

```

```

        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: STEERING
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string SUSPENSION_FRONT;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: SUSPENSION_FRONT
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute string THIRD;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL

```

```

        cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: NULL
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
            wrDb2Renau.STENE_MODEL: NULL
            wrDb2Renau.STENE_TRANDSUS: NULL
            wrDb2Renau.STENE_VERSION: NULL
            wrDb2Volks.STENE__CAR: NULL
            wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
            wrDb2Volks.STENE__GEAR: THIRD
            wrDb2Volks.STENE__MODEL: NULL
            wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string TRANS_GEAR_RATIO;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: TRANS_GEAR_RATIO
    wrDb2Volks.STENE__FUEL_CONSUMPTION: TRANS_GEAR_RATIO
        wrDb2Volks.STENE__GEAR: TRANS_GEAR_RATIO
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: TRANS_GEAR_RATIO
    )
attribute integer TRUNK_VOL_DM3;
mapping_rules(

```

```

        cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: TRUNK_VOL_DM3
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer TURNING_CIRCLE_C2C_M;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: TURNING_CIRCLE_C2C_M
        wrDb2Volks.STENE__WEIGHTS: NULL

```

```

    )
attribute integer TURNS_LOCK2LOCK;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: TURNS_LOCK2LOCK
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string TYPECHANGE;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: TYPECHANGE
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL

```

```

        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string TYPE_OF_INJECTION;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: TYPE_OF_INJECTION
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute integer VEL_II;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: VEL_II
)

```



```

        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer VEL_III;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: VEL_III
        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute integer VOL_FRONT_DM3;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL

```

```

        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: VOL_FRONT_DM3
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer VOL_REAR_DM3;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: VOL_REAR_DM3
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string WHEELS1;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL

```

```

        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: WHEELS
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: WHEELS
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer WIDTH;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: WIDTH
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer WIDTH1;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL

```

```

        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: WIDTH
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer W_AUTHORIZED_MAX_KG;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: W_AUTHORIZED_MAX_KG
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute integer W_EMPTY_MARCH_ORDER_KG;
mapping_rules(
    cardef_xml.braking_system: NULL

```

```

        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: W_EMPTY_MARCH_ORDER_KG
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute integer W_PAYLOAD_KG;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: W_PAYLOAD_KG
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )

```

```
attribute integer W_REFRAINED_TOWING_KG;
  mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: W_REFRAINED_TOWING_KG
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
  )
attribute integer W_TRAVELLING_TOTAL_KG;
  mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: W_TRAVELLING_TOTAL_KG
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
```

```

        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string b_front_mm;
mapping_rules(
    cardef_xml.braking_system: b_front_mm
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute complex braking_system;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: braking_system
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL

```

```

        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string combined;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: combined
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: MIXED_CICLE
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute complex dimensions;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: dimensions
        cardef_xml.dimensions: height_m
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: HEIGHT

```



```

        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: HEIGHT
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string drive;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: drive
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: DRIVE_TRAINS
wrDb2Volks.STENE__FUEL_CONSUMPTION: DRIVE_TRAINS
        wrDb2Volks.STENE__GEAR: DRIVE_TRAINS
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: DRIVE_TRAINS
    )
attribute complex elect_equipment;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: elect_equipment
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL

```

```

        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute complex engine;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: engine
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute string final_drive_ratio;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: final_drive_ratio

```

```

        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string fuel_cap_l;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: fuel_cap_l
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: ABILITY_TANK_L
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute complex fuel_consumption;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: fuel_consumption

```

```

        cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: NULL
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
            wrDb2Renau.STENE_MODEL: NULL
            wrDb2Renau.STENE_TRANDSUS: NULL
            wrDb2Renau.STENE_VERSION: NULL
            wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
    wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string g2;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
            cardef_xml.dimensions: NULL
            cardef_xml.fuel_consumption: NULL
                cardef_xml.gear: g2
                cardef_xml.name: NULL
                cardef_xml.speed: NULL
                cardef_xml.suspension: NULL
                cardef_xml.transmission: NULL
            cardef_xml.weights_capacities: NULL
                wrDb2Renau.STENE_MODEL: NULL
                wrDb2Renau.STENE_TRANDSUS: NULL
                wrDb2Renau.STENE_VERSION: NULL
                wrDb2Volks.STENE___CAR: NULL
            wrDb2Volks.STENE___DRIVE_TRAIN: NULL
        wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
            wrDb2Volks.STENE___GEAR: NULL
            wrDb2Volks.STENE___MODEL: NULL
            wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string g3;

```



```
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string g5;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: g5
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: VEL_V
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: FIFTH
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute string g6;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: g6
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
```

```

wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute complex gear;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: g1
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: VEL_I
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: FIRST
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL
)
attribute string kerb_weight_kg;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: kerb_weight_kg
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL

```

```

        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string length_m;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: length_m
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string luggage_capacity_dm3;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: luggage_capacity_dm3
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL

```



```

    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: VOLBAG_DM3
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute complex name;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: name
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: PCDATA_NODE
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NAME
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NAMEMODEL
        wrDb2Volks.STENE___CAR: NAME
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NAME
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NAME
        wrDb2Volks.STENE___WEIGHTS: NAME
    )
attribute string name_id;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: name_id

```

```

        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string outoftown;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: outoftown
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: CITY_EXTRA_CICLE
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: HIGHWAY_KM_L
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute complex performance;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: performance
    cardef_xml.dimensions: NULL

```

```

    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string price;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: price
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: PRICE
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: PRICE
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string rm;
mapping_rules(

```

```

        cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: rm
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string s_rear;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: s_rear
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL

```

```

    )
attribute complex speed;
  mapping_rules(
    cardef_xml.braking_system: NULL
      cardef_xml.car: NULL
      cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
      cardef_xml.gear: NULL
      cardef_xml.name: NULL
      cardef_xml.speed: NULL
      cardef_xml.suspension: NULL
      cardef_xml.transmission: speed
    cardef_xml.weights_capacities: NULL
      wrDb2Renau.STENE_MODEL: NULL
      wrDb2Renau.STENE_TRANDSUS: NULL
      wrDb2Renau.STENE_VERSION: NULL
      wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
  wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
      wrDb2Volks.STENE___GEAR: NULL
      wrDb2Volks.STENE___MODEL: NULL
      wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute complex steering;
  mapping_rules(
    cardef_xml.braking_system: NULL
      cardef_xml.car: steering
      cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
      cardef_xml.gear: NULL
      cardef_xml.name: NULL
      cardef_xml.speed: NULL
      cardef_xml.suspension: NULL
      cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
      wrDb2Renau.STENE_MODEL: NULL
      wrDb2Renau.STENE_TRANDSUS: NULL
      wrDb2Renau.STENE_VERSION: NULL
      wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
  wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL

```

```

        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute complex suspension;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: suspension
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string tires;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: tires
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: TIRES

```

```

        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: TIRES
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string towable_weight_kg;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: towable_weight_kg
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: W_NOT_REFRAINED_TOWING_KG
        wrDb2Volks.STENE___CAR: NULL
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string track_front_mm;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: track_front_mm
    cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL

```

```

        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
        wrDb2Volks.STENE__DRIVE_TRAIN: NULL
        wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string track_rear_mm;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: NULL
    cardef_xml.dimensions: track_rear_mm
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL
    cardef_xml.suspension: NULL
    cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
    wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE__GEAR: NULL
    wrDb2Volks.STENE__MODEL: NULL
    wrDb2Volks.STENE__WEIGHTS: NULL
)
attribute complex transmission;
mapping_rules(
    cardef_xml.braking_system: NULL
    cardef_xml.car: transmission
    cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: NULL
    cardef_xml.gear: NULL
    cardef_xml.name: NULL
    cardef_xml.speed: NULL

```



```

        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string urban;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL
    cardef_xml.fuel_consumption: urban
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
    cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: CITY_CICLE
        wrDb2Volks.STENE__CAR: NULL
    wrDb2Volks.STENE__DRIVE_TRAIN: NULL
wrDb2Volks.STENE__FUEL_CONSUMPTION: CITY_KM_L
        wrDb2Volks.STENE__GEAR: NULL
        wrDb2Volks.STENE__MODEL: NULL
        wrDb2Volks.STENE__WEIGHTS: NULL
    cardef_xml.braking_system:
    )
attribute string urban_range_km;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
        cardef_xml.dimensions: NULL

```

```

        cardef_xml.fuel_consumption: urban_range_km
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: NULL
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
            wrDb2Renau.STENE_MODEL: NULL
            wrDb2Renau.STENE_TRANDSUS: NULL
            wrDb2Renau.STENE_VERSION: NULL
            wrDb2Volks.STENE__CAR: NULL
            wrDb2Volks.STENE__DRIVE_TRAIN: NULL
            wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
            wrDb2Volks.STENE__GEAR: NULL
            wrDb2Volks.STENE__MODEL: NULL
            wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute complex weights_capacities;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: weights_capacities
        cardef_xml.dimensions: NULL
        cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: NULL
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
        cardef_xml.weights_capacities: NULL
            wrDb2Renau.STENE_MODEL: NULL
            wrDb2Renau.STENE_TRANDSUS: NULL
            wrDb2Renau.STENE_VERSION: NULL
            wrDb2Volks.STENE__CAR: NULL
            wrDb2Volks.STENE__DRIVE_TRAIN: NULL
            wrDb2Volks.STENE__FUEL_CONSUMPTION: NULL
            wrDb2Volks.STENE__GEAR: NULL
            wrDb2Volks.STENE__MODEL: NULL
            wrDb2Volks.STENE__WEIGHTS: NULL
    )
attribute string wheelbase_m;
mapping_rules(

```

```

        cardef_xml.braking_system: NULL
            cardef_xml.car: NULL
                cardef_xml.dimensions: wheelbase_m
cardef_xml.fuel_consumption: NULL
            cardef_xml.gear: NULL
            cardef_xml.name: NULL
            cardef_xml.speed: NULL
            cardef_xml.suspension: NULL
            cardef_xml.transmission: NULL
cardef_xml.weights_capacities: NULL
        wrDb2Renau.STENE_MODEL: NULL
        wrDb2Renau.STENE_TRANDSUS: NULL
        wrDb2Renau.STENE_VERSION: NULL
        wrDb2Volks.STENE___CAR: WHEELBASE
        wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
        wrDb2Volks.STENE___GEAR: NULL
        wrDb2Volks.STENE___MODEL: NULL
        wrDb2Volks.STENE___WEIGHTS: NULL
    )
attribute string width_m;
mapping_rules(
    cardef_xml.braking_system: NULL
        cardef_xml.car: NULL
            cardef_xml.dimensions: width_m
cardef_xml.fuel_consumption: NULL
        cardef_xml.gear: NULL
        cardef_xml.name: NULL
        cardef_xml.speed: NULL
        cardef_xml.suspension: NULL
        cardef_xml.transmission: NULL
cardef_xml.weights_capacities: NULL
    wrDb2Renau.STENE_MODEL: NULL
    wrDb2Renau.STENE_TRANDSUS: NULL
    wrDb2Renau.STENE_VERSION: NULL
    wrDb2Volks.STENE___CAR: NULL
    wrDb2Volks.STENE___DRIVE_TRAIN: NULL
wrDb2Volks.STENE___FUEL_CONSUMPTION: NULL
    wrDb2Volks.STENE___GEAR: NULL
    wrDb2Volks.STENE___MODEL: NULL
    wrDb2Volks.STENE___WEIGHTS: NULL

```

```
        )
    }
interface Change
{
    attribute string CODC;
        mapping_rules(
            wrDb2Renau.STENE_CHANGE: CODC
        )
    attribute string TYPE;
        mapping_rules(
            wrDb2Renau.STENE_CHANGE: TYPE
        )
}
interface Cluster0
{
    attribute array car;
        mapping_rules(
            cardef_xml.fiat: car
        )
}
interface Cluster1
{
    attribute complex e_electrical;
        mapping_rules(
            cardef_xml.engine: e_electrical
        )
    attribute complex e_fuel;
        mapping_rules(
            cardef_xml.engine: e_fuel
        )
}
interface Cluster2
{
    attribute complex p_coupe;
        mapping_rules(
            cardef_xml.performance: p_coupe
        )
    attribute complex p_electrical;
        mapping_rules(
            cardef_xml.performance: p_electrical
        )
}
```

```

attribute complex p_methane;
    mapping_rules(
        cardef_xml.performance: p_methane
    )
attribute complex p_other;
    mapping_rules(
        cardef_xml.performance: p_other
    )
}
interface Electrical_system
{
    attribute string ALTERNATOR_V_A;
        mapping_rules(
            cardef_xml.elect_equipment: NULL
wrDb2Volks.STENE___ELECTRICAL_SYSTEM: ALTERNATOR_V_A
        )
    attribute integer BATTERY_V;
        mapping_rules(
            cardef_xml.elect_equipment: NULL
wrDb2Volks.STENE___ELECTRICAL_SYSTEM: BATTERY_V
        )
    attribute string IGNITION;
        mapping_rules(
            cardef_xml.elect_equipment: NULL
wrDb2Volks.STENE___ELECTRICAL_SYSTEM: IGNITION
        )
    attribute string alternator_current;
        mapping_rules(
            cardef_xml.elect_equipment: alternator_current
wrDb2Volks.STENE___ELECTRICAL_SYSTEM: NULL
        )
    attribute string battery_capacity;
        mapping_rules(
            cardef_xml.elect_equipment: battery_capacity
wrDb2Volks.STENE___ELECTRICAL_SYSTEM: BATTERY_A_HRS
        )
}
interface Engine
{
    attribute string ALTERNATOR_V_A;
        mapping_rules(

```

```

        cardef_xml.e_electrical: NULL
        cardef_xml.e_fuel: NULL
        wrDb2Renau.STENE_MOTOR: NULL
        wrDb2Volks.STENE___ENGINE: ALTERNATOR_V_A
    )
attribute string ARRANGEMENT;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: ARRANGEMENT
)
attribute integer BATTERY_A_HRS;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: BATTERY_A_HRS
)
attribute integer BATTERY_V;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: BATTERY_V
)
attribute string COOLING_SYSTEM;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: COOLING_SYSTEM
)
attribute integer COOLING_SYSTEM_L;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: COOLING_SYSTEM_L
)
attribute string CRANK_SHAFT;

```

```
mapping_rules(  
    cardef_xml.e_electrical: NULL  
        cardef_xml.e_fuel: NULL  
        wrDb2Renau.STENE_MOTOR: NULL  
        wrDb2Volks.STENE___ENGINE: CRANK_SHAFT  
    )  
attribute string CYLINDER_BLOCK;  
mapping_rules(  
    cardef_xml.e_electrical: NULL  
        cardef_xml.e_fuel: NULL  
        wrDb2Renau.STENE_MOTOR: NULL  
        wrDb2Volks.STENE___ENGINE: CYLINDER_BLOCK  
    )  
attribute string CYLINDER_HEAD;  
mapping_rules(  
    cardef_xml.e_electrical: NULL  
        cardef_xml.e_fuel: NULL  
        wrDb2Renau.STENE_MOTOR: NULL  
        wrDb2Volks.STENE___ENGINE: CYLINDER_HEAD  
    )  
attribute string EMISSION;  
mapping_rules(  
    cardef_xml.e_electrical: NULL  
        cardef_xml.e_fuel: NULL  
        wrDb2Renau.STENE_MOTOR: NULL  
        wrDb2Volks.STENE___ENGINE: EMISSION  
    )  
attribute integer ENGINE_OIL_L;  
mapping_rules(  
    cardef_xml.e_electrical: NULL  
        cardef_xml.e_fuel: NULL  
        wrDb2Renau.STENE_MOTOR: NULL  
        wrDb2Volks.STENE___ENGINE: ENGINE_OIL_L  
    )  
attribute string FIRE_ORDER;  
mapping_rules(  
    cardef_xml.e_electrical: NULL  
        cardef_xml.e_fuel: NULL  
        wrDb2Renau.STENE_MOTOR: NULL  
        wrDb2Volks.STENE___ENGINE: FIRE_ORDER  
    )
```

```
attribute string FUEL_AIR_SUPPLY;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: FUEL_AIR_SUPPLY
)
attribute integer FUEL_TANK_L;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: FUEL_TANK_L
)
attribute string LUBRIFICATION;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: LUBRIFICATION
)
attribute integer STROKE_MM;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: STROKE_MM
)
attribute integer WIPER_FLUID_L;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: WIPER_FLUID_L
)
attribute string bat_charge_time;
mapping_rules(
    cardef_xml.e_electrical: bat_charge_time
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: NULL
)
```



```

    )
attribute string bore_mm;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: bore_mm
    wrDb2Renau.STENE_MOTOR: BORE_X_RACE_MM
    wrDb2Volks.STENE___ENGINE: BORE_MM
)
attribute string capacity_cc;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: capacity_cc
    wrDb2Renau.STENE_MOTOR: PISTON_DISPLACEMENT_CC
    wrDb2Volks.STENE___ENGINE: DISPLACEMENT_CC
)
attribute string compression_ratio;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: compression_ratio
    wrDb2Renau.STENE_MOTOR: COMPRESSION_RATIO
    wrDb2Volks.STENE___ENGINE: COMPRESSION_RATIO
)
attribute string cylinders;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: cylinders
    wrDb2Renau.STENE_MOTOR: NO_OF_CYLINDER
    wrDb2Volks.STENE___ENGINE: NULL
)
attribute string drive_battery_kwh;
mapping_rules(
    cardef_xml.e_electrical: drive_battery_kwh
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: NULL
)
attribute string e_name1;
mapping_rules(
    cardef_xml.e_electrical: e_name
    cardef_xml.e_fuel: e_name
    wrDb2Renau.STENE_MOTOR: COD

```

```
        wrDb2Volks.STENE___ENGINE: COD_E
    )
attribute string e_type;
mapping_rules(
    cardef_xml.e_electrical: e_type
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: TYPE
)
attribute string engine_control;
mapping_rules(
    cardef_xml.e_electrical: engine_control
    cardef_xml.e_fuel: NULL
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: NULL
)
attribute string fuel_system;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: fuel_system
    wrDb2Renau.STENE_MOTOR: FUEL
    wrDb2Volks.STENE___ENGINE: FUEL_REQUIREMENT
)
attribute string ignition;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: ignition
    wrDb2Renau.STENE_MOTOR: TYPE_OF_INJECTION
    wrDb2Volks.STENE___ENGINE: IGNITION
)
attribute string layout1;
mapping_rules(
    cardef_xml.e_electrical: layout
    cardef_xml.e_fuel: layout
    wrDb2Renau.STENE_MOTOR: NULL
    wrDb2Volks.STENE___ENGINE: NULL
)
attribute string power_bhp;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: power_bhp
```

```

        wrDb2Renau.STENE_MOTOR: POWER_CV
        wrDb2Volks.STENE___ENGINE: HORSEPOWER
    )
attribute string power_kw1;
mapping_rules(
    cardef_xml.e_electrical: power_kw
    cardef_xml.e_fuel: power_kw
    wrDb2Renau.STENE_MOTOR: POWER_KW
    wrDb2Volks.STENE___ENGINE: KW
)
attribute string torque_kgm;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: torque_kgm
    wrDb2Renau.STENE_MOTOR: BRACE_MKG
    wrDb2Volks.STENE___ENGINE: MAX_TORQUE_MKG
)
attribute string torque_nml;
mapping_rules(
    cardef_xml.e_electrical: torque_nm
    cardef_xml.e_fuel: torque_nm
    wrDb2Renau.STENE_MOTOR: BRACE_NM
    wrDb2Volks.STENE___ENGINE: MAX_TORQUE_NM
)
attribute string valve_gear;
mapping_rules(
    cardef_xml.e_electrical: NULL
    cardef_xml.e_fuel: valve_gear
    wrDb2Renau.STENE_MOTOR: NO_OF_VALVES
    wrDb2Volks.STENE___ENGINE: VALVE_TRAIN
)
}
interface Performance
{
attribute string p_0_1000m2;
mapping_rules(
    cardef_xml.p_coupe: p_0_1000m
cardef_xml.p_electrical: NULL
    cardef_xml.p_methane: p_0_1000m
    cardef_xml.p_other: p_0_1000m
)
}

```

```
attribute string p_0_100_km_h2;
    mapping_rules(
        cardef_xml.p_coupe: p_0_100_km_h
        cardef_xml.p_electrical: NULL
        cardef_xml.p_methane: p_0_100_km_h
        cardef_xml.p_other: p_0_100_km_h
    )
attribute string p_0_50_km_h;
    mapping_rules(
        cardef_xml.p_coupe: NULL
        cardef_xml.p_electrical: p_0_50_km_h
        cardef_xml.p_methane: NULL
        cardef_xml.p_other: NULL
    )
attribute string pickup_1000m_sec;
    mapping_rules(
        cardef_xml.p_coupe: pickup_1000m_sec
        cardef_xml.p_electrical: NULL
        cardef_xml.p_methane: NULL
        cardef_xml.p_other: NULL
    )
attribute string range_methane_km;
    mapping_rules(
        cardef_xml.p_coupe: NULL
        cardef_xml.p_electrical: NULL
        cardef_xml.p_methane: range_methane_km
        cardef_xml.p_other: NULL
    )
attribute string top_speed3;
    mapping_rules(
        cardef_xml.p_coupe: top_speed
        cardef_xml.p_electrical: top_speed
        cardef_xml.p_methane: top_speed
        cardef_xml.p_other: top_speed
    )
}
interface Steering
{
attribute string STEERING;
    mapping_rules(
        cardef_xml.steering: NULL
    )
}
```

```
wrDb2Volks.STENE__STEERING: STEERING
    )
attribute string s_type;
    mapping_rules(
        cardef_xml.steering: s_type
wrDb2Volks.STENE__STEERING: NULL
    )
attribute string turning_circle;
    mapping_rules(
        cardef_xml.steering: turning_circle
wrDb2Volks.STENE__STEERING: NULL
    )
}
```

5.5 Conclusioni

Il presente progetto ha portato alla costruzione di uno schema globale in cui sono stati integrati i dati conservati in sorgenti strutturate e semistrutturate relativi alle auto della Renault, della Volkswagen e della Fiat.

Come abbiamo osservato il processo di integrazione, gestito tramite il sistema MOMIS, è stato realizzato quasi completamente in automatico, impegnando poco il progettista se non nella scelta delle forme base e dei significati dei termini che identificavano le classi e gli attributi delle sorgenti.

Da ciò si può desumere che il sistema MOMIS è uno strumento potente per gestire dati eterogenei e supplire alle carenze evidenziate, al capitolo due, nel commercio elettronico.

Ovviamente la realizzazione di e-commerce che sfrutti l'integrazione di diverse sorgenti non può fermarsi alla costruzione dello schema globale, ma esige anche una interfaccia web che invocando il Query Manager possa accedere alle sorgenti locali e fornire ai clienti le informazioni desiderate; questo sarà oggetto di una futura tesi.

Appendice A

Il linguaggio ODL³

Si propone la descrizione BNF del linguaggio ODL_{I³}. Sono stati inclusi unicamente gli elementi sintattici che differiscono dalla grammatica originale ODL dello standard ODMG-93.

```
⟨interface_dcl⟩ ::= ⟨interface_header⟩
                  { [⟨ interface_body ⟩ ] };
                  [ union ⟨ identifier ⟩ { ⟨ interface_body ⟩ } ; ]
⟨interface_header⟩ ::= interface ⟨ identifier ⟩ [⟨ inheritance_spec ⟩]
                    [⟨ type_property_list ⟩]
⟨ inheritance_spec ⟩ ::= : ⟨ scoped_name ⟩
                      [, ⟨ inheritance_spec ⟩]
```

Definizione di modello di schema locale: il wrapper deve potere indicare il tipo e il nome della sorgente per ogni modello.

```

⟨type_property_list⟩ ::= ( [⟨source_spec⟩]
                          [⟨extent_spec⟩]
                          [⟨key_spec⟩] [⟨f_key_spec⟩] [⟨c_key_spec⟩] )
⟨source_spec⟩       ::= source ⟨source_type⟩
                          ⟨source_name⟩
⟨source_type⟩       ::= relational | nfrelational
                          | object | file
                          | semistructured
⟨source_name⟩       ::= ⟨identifier⟩
⟨extent_spec⟩       ::= extent ⟨extent_list⟩
⟨extent_list⟩       ::= ⟨string⟩ | ⟨string⟩,⟨extent_list⟩
⟨key_spec⟩          ::= key[s] ⟨key_list⟩
⟨f_key_spec⟩        ::= foreign_key (⟨f_key_list⟩)
                          references ⟨key_list
                          ) [⟨f_key_spec⟩]
⟨c_key_spec⟩        ::= candidate_key ⟨identifier⟩
                          (⟨key_list⟩)

```

Regole di definizione del mapping fra attributi della classe globale dello schema del mediatore e i corrispondenti nelle sorgenti locali.

```

⟨attr_dcl⟩          ::= [readonly] attribute
                          [⟨domain_type⟩]
                          ⟨attribute_name⟩ [*]
                          [⟨fixed_array_size⟩]
                          [⟨mapping_rule_dcl⟩]
⟨mapping_rule_dcl⟩ ::= mapping_rule ⟨rule_list⟩
⟨rule_list⟩         ::= ⟨rule⟩ | ⟨rule⟩,⟨rule_list⟩
⟨rule⟩              ::= ⟨local_attr_name⟩ |
                          ‘⟨identifier⟩’
                          ⟨and_expression⟩ |
                          ⟨union_expression⟩
⟨and_expression⟩    ::= ( ⟨local_attr_name⟩ and
                          ⟨and_list⟩ )
⟨and_list⟩          ::= ⟨local_attr_name⟩
                          | ⟨local_attr_name⟩ and
                          ⟨and_list⟩
⟨union_expression⟩ ::= ( ⟨local_attr_name⟩ union
                          ⟨union_list⟩ on ⟨identifier⟩ )
⟨union_list⟩        ::= ⟨local_attr_name⟩

```

| $\langle \text{local_attr_name} \rangle$ **union**
 $\langle \text{union_list} \rangle$

$\langle \text{local_attr_name} \rangle ::= \langle \text{source_name} \rangle . \langle \text{class_name} \rangle .$
 $\langle \text{attribute_name} \rangle$

...

Relazioni terminologiche utilizzate per definire il Common Thesaurus.

$\langle \text{relationships_list} \rangle ::= \langle \text{relationship_dcl} \rangle ; |$
 $\langle \text{relationship_dcl} \rangle ;$
 $\langle \text{relationships_list} \rangle$

$\langle \text{relationships_dcl} \rangle ::= \langle \text{local_name} \rangle$
 $\langle \text{relationship_type} \rangle$
 $\langle \text{local_name} \rangle$

$\langle \text{local_name} \rangle ::= \langle \text{source_name} \rangle .$
 $\langle \text{local_class_name} \rangle$
 $[. \langle \text{local_attr_name} \rangle]$

$\langle \text{relationship_type} \rangle ::= \mathbf{SYN} | \mathbf{BT} | \mathbf{NT} | \mathbf{RT}$

...

Definizione dei vincoli di integrità **OLCD** dichiarazione delle regole (utilizzando le definizioni *if then*) valide per ogni istanza di dato; specificazione delle mapping rule (specificazione delle regole *or* e *and*).

$\langle \text{rule_list} \rangle ::= \langle \text{rule_dcl} \rangle ; | \langle \text{rule_dcl} \rangle ; \langle \text{rule_list} \rangle$

$\langle \text{rule_dcl} \rangle ::= \mathbf{rule} \langle \text{identifier} \rangle \langle \text{rule_spec} \rangle$

$\langle \text{rule_spec} \rangle ::= \langle \text{rule_pre} \rangle \mathbf{then} \langle \text{rule_post} \rangle |$
 $\{ \langle \text{case_dcl} \rangle \}$

$\langle \text{rule_pre} \rangle ::= \langle \text{forall} \rangle \langle \text{identifier} \rangle \mathbf{in} \langle \text{identifier} \rangle :$
 $\langle \text{rule_body_list} \rangle$

$\langle \text{rule_post} \rangle ::= \langle \text{rule_body_list} \rangle$

$\langle \text{case_dcl} \rangle ::= \mathbf{case\ of} \langle \text{identifier} \rangle : \langle \text{case_list} \rangle$

$\langle \text{case_list} \rangle ::= \langle \text{case_spec} \rangle | \langle \text{case_spec} \rangle \langle \text{case_list} \rangle$

$\langle \text{case_spec} \rangle ::= \langle \text{identifier} \rangle : \langle \text{identifier} \rangle ;$

$\langle \text{rule_body_list} \rangle ::= (\langle \text{rule_body_list} \rangle) |$
 $\langle \text{rule_body} \rangle |$
 $\langle \text{rule_body_list} \rangle \mathbf{and}$
 $\langle \text{rule_body} \rangle |$
 $\langle \text{rule_body_list} \rangle \mathbf{and}$
 $(\langle \text{rule_body_list} \rangle)$
 $\langle \text{rule_body} \rangle ::= \langle \text{dotted_name} \rangle$
 $\langle \text{rule_const_op} \rangle$
 $\langle \text{literal_value} \rangle |$
 $\langle \text{dotted_name} \rangle$
 $\langle \text{rule_const_op} \rangle$
 $\langle \text{rule_cast} \rangle \langle \text{literal_value} \rangle |$
 $\langle \text{dotted_name} \rangle \mathbf{in}$
 $\langle \text{dotted_name} \rangle |$
 $\langle \text{forall} \rangle \langle \text{identifier} \rangle \mathbf{in}$
 $\langle \text{dotted_name} \rangle :$
 $\langle \text{rule_body_list} \rangle |$
 $\mathbf{exists} \langle \text{identifier} \rangle \mathbf{in}$
 $\langle \text{dotted_name} \rangle :$
 $\langle \text{rule_body_list} \rangle$
 $\langle \text{rule_const_op} \rangle ::= = | \geq | \leq | > | <$
 $\langle \text{rule_cast} \rangle ::= (\langle \text{simple_type_spec} \rangle)$
 $\langle \text{dotted_name} \rangle ::= \langle \text{identifier} \rangle | \langle \text{identifier} \rangle .$
 $\langle \text{dotted_name} \rangle$
 $\langle \text{forall} \rangle ::= \mathbf{for\ all} | \mathbf{forall}$

Bibliografia

- [1] Choi et al. *The Economics of Electronic Commerce*. MacMillan, 1997.
- [2] Aris Ouksel. Lecture on e-commerce.
<http://155.185.21.37/bdati/Seminario99-00.html>
- [3] T. Berners-Lee/CN, R. Cailliau/ECP. WorldWideWeb: Proposal for a HyperText Project. <http://www.w3.org/History/19921103-hypertext/hypertext/WWW/Proposal.html>, 12 November 1990
- [4] World Wide Web Consortium. Some early ideas for HTML.
<http://www.w3.org/MarkUp/#historical>.
- [5] World Wide Web Consortium. Extensible Markup Language (xml) 1.0.
<http://www.w3.org/TR/REC-xml>, 1998.
- [6] Dan Rogers. The BizTalk Philosophy.
<http://www.biztalk.org/Biztalk/philosophy.asp>, 2000.
- [7] Gio Wiederhold et al. Integrating Artificial Intelligence and Database Technology. *Journal of Intelligent Information Systems*, volume 2/3, June 1996.
- [8] R. Hull and R. King et al. Arpa i³ reference architecture, 1995.
http://www.isse.gmu.edu/I3_Arch/index.html.
- [9] G.Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, volume 25, pages 38–49, 1992.
- [10] N.Guarino. Semantic matching: Formal ontological distinctions for information organization, extraction, and integration. Technical report, Summer School on Information Extraction, Frascati, Italy, July 1997.
- [11] N.Guarino. Understanding, building, and using ontologies. A commentary to 'Using Explicit Ontologies in KBS Development', by van Heijst, Schreiber, and Wielinga.

- [12] F. Saltor and E. Rodriguez. On intelligent access to heterogeneous information. In *Proceedings of the 4th KRDB Workshop*, Athens, Greece, August 1997.
- [13] Arthur M. Keller. Smart Catalogs and Virtual Catalogs. In *International Conference on Frontiers of Electronic Commerce*, October 1995. Available at <http://www-db.stanford.edu/pub/keller/>
- [14] S. Bergamaschi, C. Sartori, M. Vincini and D. Beneventano. Semantic Integration of Heterogeneous Information Sources. *Journal of Data and Knowledge Engineering*, 1999.
- [15] S. Bergamaschi, D. Beneventano, S. Castano and M. Vincini. Integrazione di informazione: il linguaggio ODLI3 e la logica descrittiva OLCD. Technical report T3-R03, 16 Luglio 1998.
- [16] A. Zaccaria. Momis: Il componente query manager. Tesi di Laurea, Università di Modena, Facoltà di Ingegneria, corso di laurea in Ingegneria Informatica, 1998.
- [17] M. Franceschi. Il componente query manager di momis: utilizzo della conoscenza estensionale. Tesi di Laurea, Università di Modena e Reggio Emilia, Facoltà di Ingegneria, corso di laurea in Ingegneria Informatica, 2000.
- [18] R. G. G. Cattell. The Object Database Standard: ODMG93. Morgan Kaufmann Publishers, San Mateo, CA, 1994.
- [19] R.G.G. Cattell and others. The Object Data Standard: ODMG 2.0, Morgan Kaufmann Publishers, San Francisco, CA, 1997.
- [20] D. Beneventano, S. Bergamaschi, C. Sartori, and M. Vincini. Odb-tools: a description logics based tool for schema validation and semantic query optimization in object oriented databases. In *Sesto Convegno AIIA - Roma*, 1997.
- [21] Domenico Beneventano, Sonia Bergamaschi, Claudio Sartori, and Maurizio Vincini. ODB-QOPTIMIZER: A tool for semantic query optimization in oodb. In *Int. Conference on Data Engineering - ICDE97*, 1997. <http://sparc20.dsi.unimo.it>.
- [22] A.G. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

- [23] D. Beneventano, S. Bergamaschi, A. Corni, R. Guidetti, G. Malvezzi SI-Designer un tool di ausilio all'integrazione intelligente di sorgenti di informazione. In *SEBD: Sistemi Evoluti per Basi di Dati*, Palazzo dell'Emiciclo L'Aquila, Italy, June 2000.
- [24] T. Catarci and M. Lenzerini. Representing and using interschema knowledge in cooperative information systems. *Journal of Intelligent and Cooperative Information Systems*, volume 2/4, pages 375-398, 1993.
- [25] R. Guidetti. SI-Designer: un tool per l'integrazione di sorgenti distribuite ed eterogenee Tesi di Laurea, Università di Modena, Facoltà di Ingegneria, corso di laurea in Ingegneria Informatica, 2000.
- [26] The MOMIS Project. <http://sparc20.dsi.unimo.it/Momis/>
- [27] G. Malvezzi. Estrazione di relazioni lessicali con WordNet nel sistema MOMIS Tesi di Laurea, Università di Modena, Facoltà di Ingegneria, corso di laurea in Ingegneria Informatica, 2000.
- [28] F. Venuta. Trattamento della Conoscenza Estensionale nel sistema MOMIS Tesi di Laurea, Università di Modena, Facoltà di Ingegneria, corso di laurea in Ingegneria Informatica, 2000.