

# Logiche descrittive per l'integrazione di dati e servizi

*Maurizio Lenzerini*

**Dipartimento di Informatica e Sistemistica “Antonio Ruberti”  
Università di Roma “La Sapienza”**

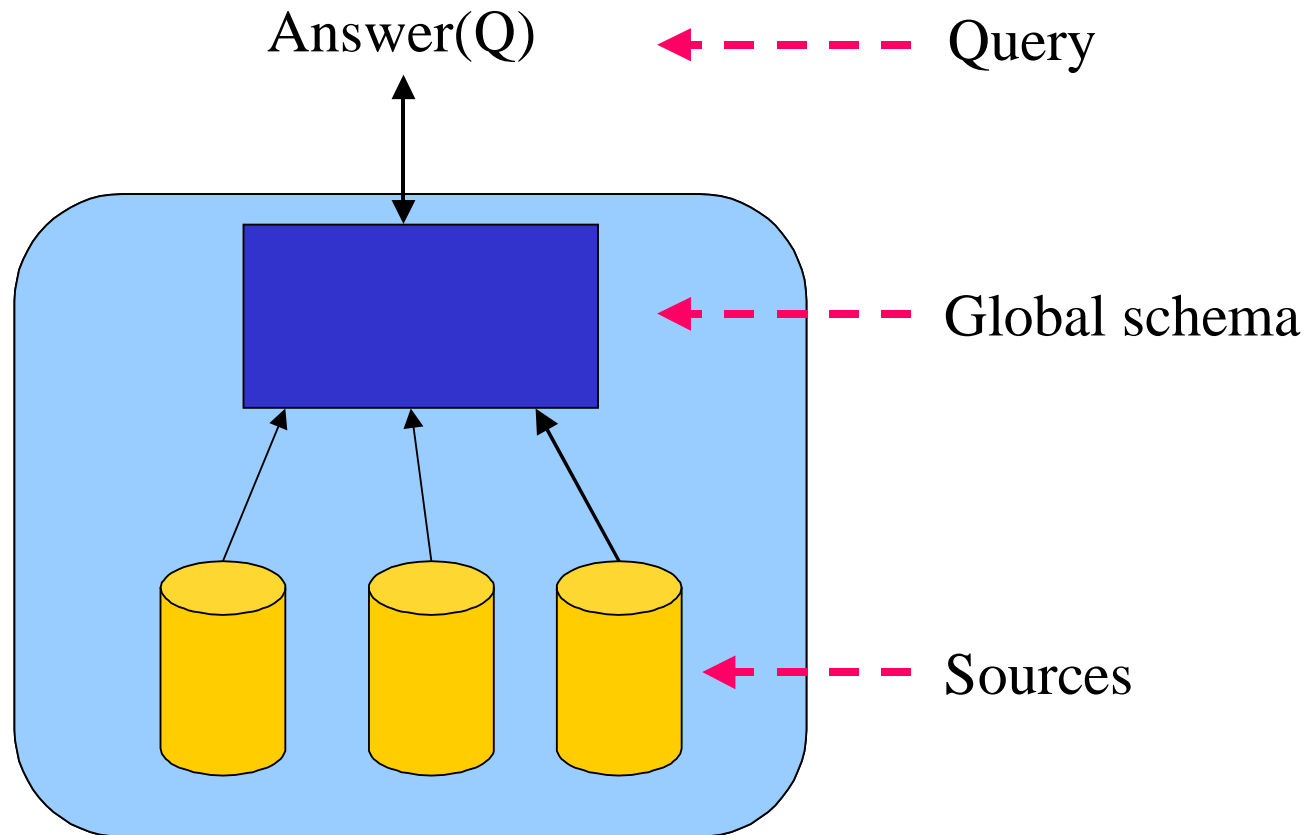
**I sistemi informativi basati sulle logiche descrittive**

*FORUM P.A. – Roma, 24 novembre 2005*

## Integrazione e cooperazione

- Attualmente una delle maggiori sfide della ICT
- Mercato in grande espansione (7.5 billion \$ – Aberdeen Group)
- Due facce dello stesso obiettivo
  - Integrazione: intra-organizzazione (e.g., EIS)
  - Cooperazione: inter-organizzazione
- Oggetto della integrazione e della cooperazione
  - Dati
  - Servizi (esigenza più recente)

## Integrazione dei dati



Meccanismi per l'accesso integrato ai dati sono cruciali in moltissimi contesti:

- sistemi informativi distribuiti, data warehouse e di gestione della conoscenza
- e-commerce, e-business, e-government,
- gestione di dati biologici e scientifici, e ... molto altro

## Integrazione dei dati: un po' di storia

Vari aspetti, approcci e architetture:

- **integrazione di schemi**

gli schemi sorgenti vengono fusi allo scopo di produrre un unico schema –  
l'integrazione **non** riguarda i dati

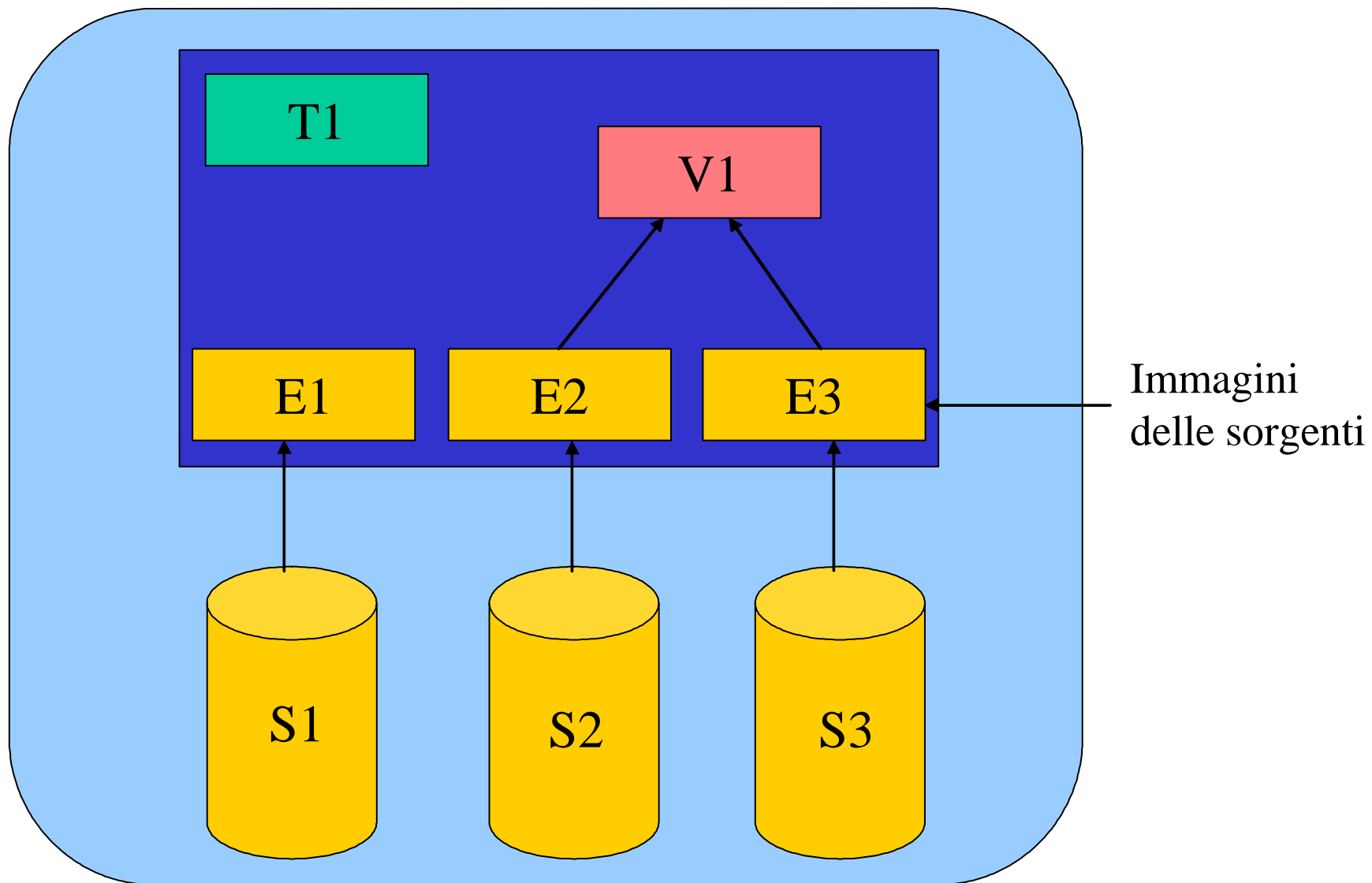
- **basi di dati distribuite**

i dati sono integrati per mezzo di un sistema di gestione di basi di dati distribuite,  
che raccoglie un insieme omogeneo di sorgenti di dati sotto il controllo di  
un'unica organizzazione

- **basi di dati federate**

le sorgenti di dati sono autonome, e l'integrazione viene realizzata con  
meccanismi impliciti (procedurali), a livello del modello logico dei dati (ad es.,  
relazionale)

# Strumenti per basi di dati federate



# Vantaggi e svantaggi degli strumenti per basi di dati federate

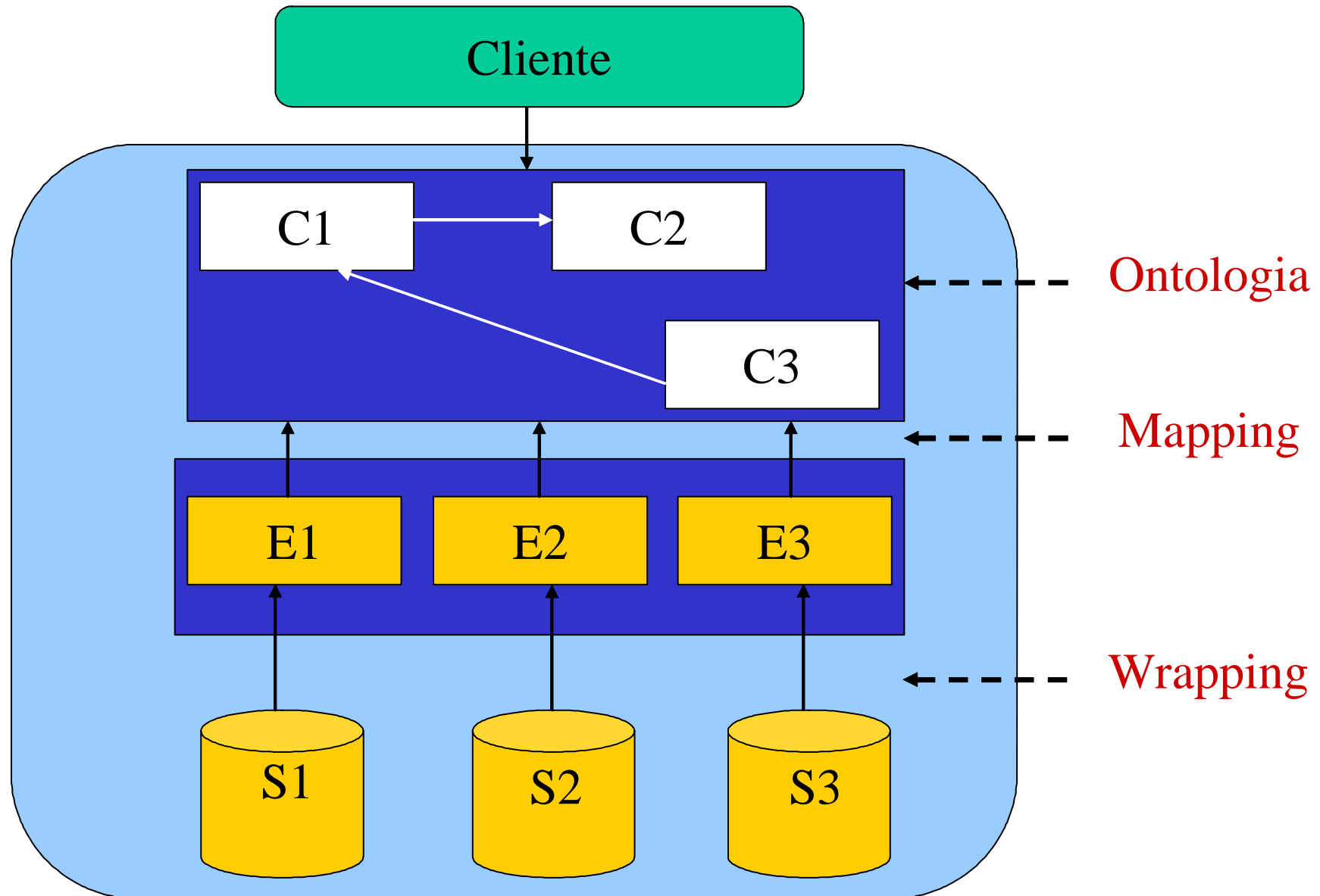
## Vantaggi

- **Trasparenza fisica**
- **Eterogeneità**
- **Estendibilità**
- **Autonomia** delle sorgenti dei dati
- **Efficienza** (ottimizzazione delle query distribuite)

## Svantaggi

- Tuttavia, gli strumenti di mercato (ad esempio, DB2II) **non** supportano la **trasparenza logica/concettuale**

# Trasparenza logica/concettuale nell'integrazione basata su mediatore



## Trasparenza logica/concettuale

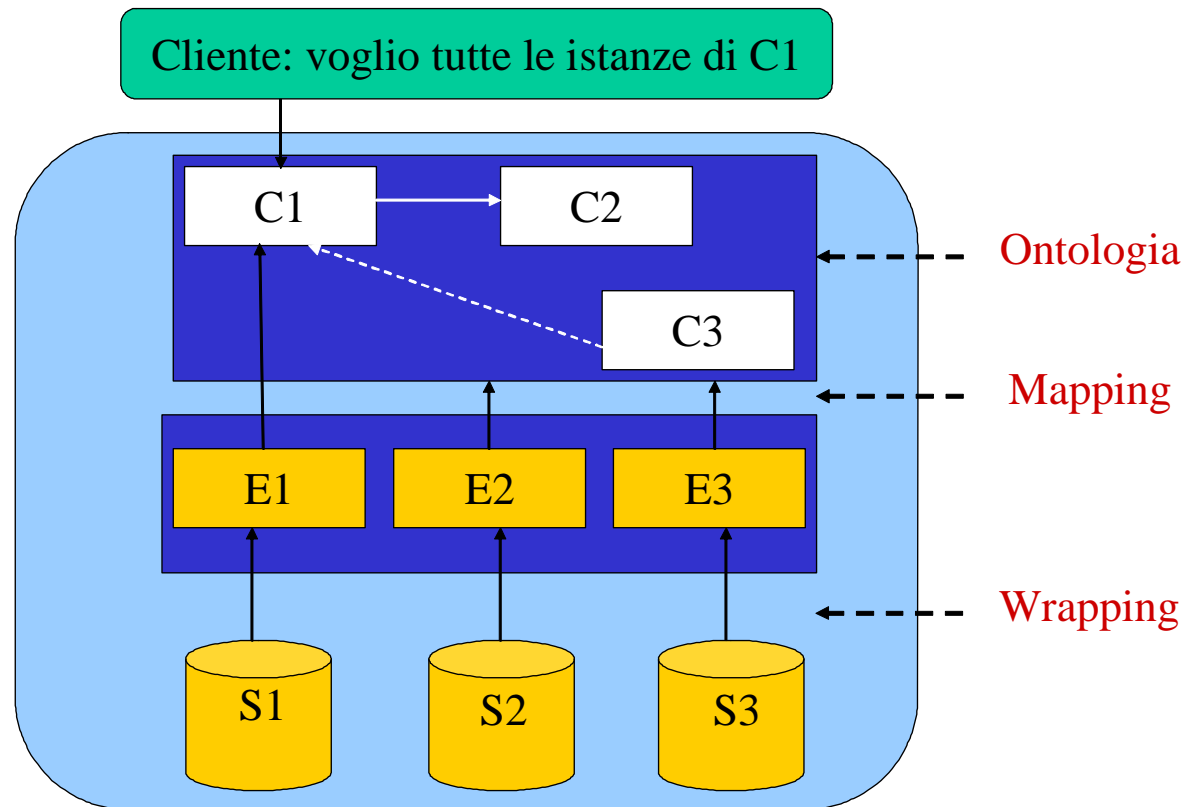
Ingredienti di base per la trasparenza logica/concettuale

- lo schema globale (ontologia) fornisce una vista concettuale indipendente dalle sorgenti
- lo schema globale viene descritto con un formalismo ricco dal punto di vista della semantica
- i mapping (ed il wrapping) sono strumenti cruciali per realizzare l'indipendenza tra schema globale e sorgenti, e stabiliscono corrispondenze tra concetti complessi dell'ontologia e query sulle sorgenti

La ricerca della trasparenza logico/concettuale è un aspetto dominante in molte aree:

- linguaggi di programmazione
- ingegneria del software
- basi di dati
- intelligenza artificiale

## Perché l'obiettivo della trasparenza logica/concettuale è fallito finora?



Supponiamo che C3 implichi logicamente C1 (linea tratteggiata). Se il sistema non è in grado di dedurre questa proprietà, per rispondere alla query accederà solo ad S1, tralasciando la sorgente S3, e non rispondendo correttamente alle query. Serve quindi uno strumento di **ragionamento sulle ontologie**.

## Ontologie e logiche descrittive

- Una **ontologia** è una specifica formale di una concettualizzazione di un dominio, che rende esplicito il significato dei concetti rilevanti nel dominio
- In Computer Science l'interesse e la ricerca sulle ontologie è iniziato in intelligenza artificiale negli anni '80, ed è ora molto rilevante in diverse aree
- Tradizionalmente, una ontologia mira ad esplicitare **classi/concetti/entità** e **relazioni** tra questi
- Per poter sfruttare appieno una ontologia, è fondamentale esprimerla in **logica**. Le **logiche descrittive** sono logiche specificamente definite per esprimere la **conoscenza** in termini di concetti e relazioni, con associate procedure automatiche di **ragionamento corretto e completo**, e caratterizzate dal punto di vista **computazionale**


## Logiche descrittive: un po' di storia

- **Anni '70**: Reti semantiche e sistemi a frame (cfr. R.J. Brachman: What's in a Concept: Structural Foundations for Semantic Networks. International Journal of Man-Machine Studies 9(2): 127-152, 1977)
- **Anni '80**: Nascono le logiche descrittive (cfr. R. J. Brachman, H. J. Levesque: The Tractability of Subsumption in Frame-Based Description Languages. AAAI 1984: 34-37)
- **Anni 1987-1995**: Tre gruppi di ricerca (DIS-UNIROMA1, Saarbruecken, Bell Lab) conducono una analisi formale e computazionale dei linguaggi usati nelle logiche descrittive (cfr. F. M. Donini, M. Lenzerini, D. Nardi, W. Nutt: The Complexity of Concept Languages. Information and Computation 134(1): 1-58, 1997)
- **Anni 1995-2000**: Analisi formale e computazionale delle deduzioni su ontologie espresse in logiche descrittive (cfr. tesi di dottorato di Giuseppe De Giacomo e Diego Calvanese - DIS-UNIROMA1)
- **Anni 2000**: Sistemi e applicazioni (cfr. The Description Logic Handbook: Theory, Implementation and Applications, Cambridge University Press, 2003.)

# Complessità del ragionamento in varie logiche descrittive

$\mathcal{AL}$ +	P	Co-NP		NP			PSPACE								
$C \sqcup D$			×	×				×	×		×	×	×		×
$(\geq n R)$ $(\leq n R)$		×		×						×	×		×	×	×
$\exists R.C$					×		×	×			×	×		×	×
$R \sqcap R'$						×	×		×	×		×	×	×	×

## Il sistema QuOnto: modellare ed interrogare ontologie

- QuOnto  è un sistema sviluppato dal Dipartimento di Informatica e Sistemistica dell'Università di Roma "La Sapienza" (DIS-UNIROMA1), in particolare dal DASI-lab (Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati)
- Consente per modellare, gestire ed interrogare ontologie in una logica descrittiva (DL-lite) risultato di circa vent'anni di ricerche
- I linguaggi utilizzati (di modeling e querying) per questo scopo si basano su logiche definite per raggiungere il compromesso ottimale tra espressività ed efficienza del ragionamento e nell'accesso a grandi quantità di dati

## Alcuni risultati di complessità

$B$	$C$	$R$	$(\text{funct } R)$	Data complexity
$A \mid \exists R \mid B_1 \sqcap B_2$	$A \mid \perp \mid \exists R$	$P \mid P^-$	<i>allowed</i>	in LOGSPACE
$A \mid \exists R \mid B_1 \sqcap B_2$	$A \mid \perp \mid \exists R.C$	$P \mid P^-$	<i>not allowed</i>	in LOGSPACE
$A \mid \exists P.A$	$A$	$P$	<i>not allowed</i>	NLOGSPACE-hard
$A$	$A \mid \forall P.A$	$P$	<i>not allowed</i>	NLOGSPACE-hard
$A$	$A \mid \exists P.A$	$P$	<i>allowed</i>	NLOGSPACE-hard
$A \mid \exists P.A \mid B_1 \sqcap B_2$	$A$	$P$	<i>not allowed</i>	P TIME-hard
$A \mid B_1 \sqcap B_2$	$A \mid \forall P.A$	$P$	<i>not allowed</i>	P TIME-hard
$A \mid B_1 \sqcap B_2$	$A \mid \exists P.A$	$P$	<i>allowed</i>	P TIME-hard
$A \mid \neg A$	$A$	$P$	<i>not allowed</i>	coNP-hard
$A$	$A \mid A_1 \sqcup A_2$	$P$	<i>not allowed</i>	coNP-hard
$A \mid \forall P.A$	$A$	$P$	<i>not allowed</i>	coNP-hard

## Il sistema QuOnto per l'integrazione dei dati

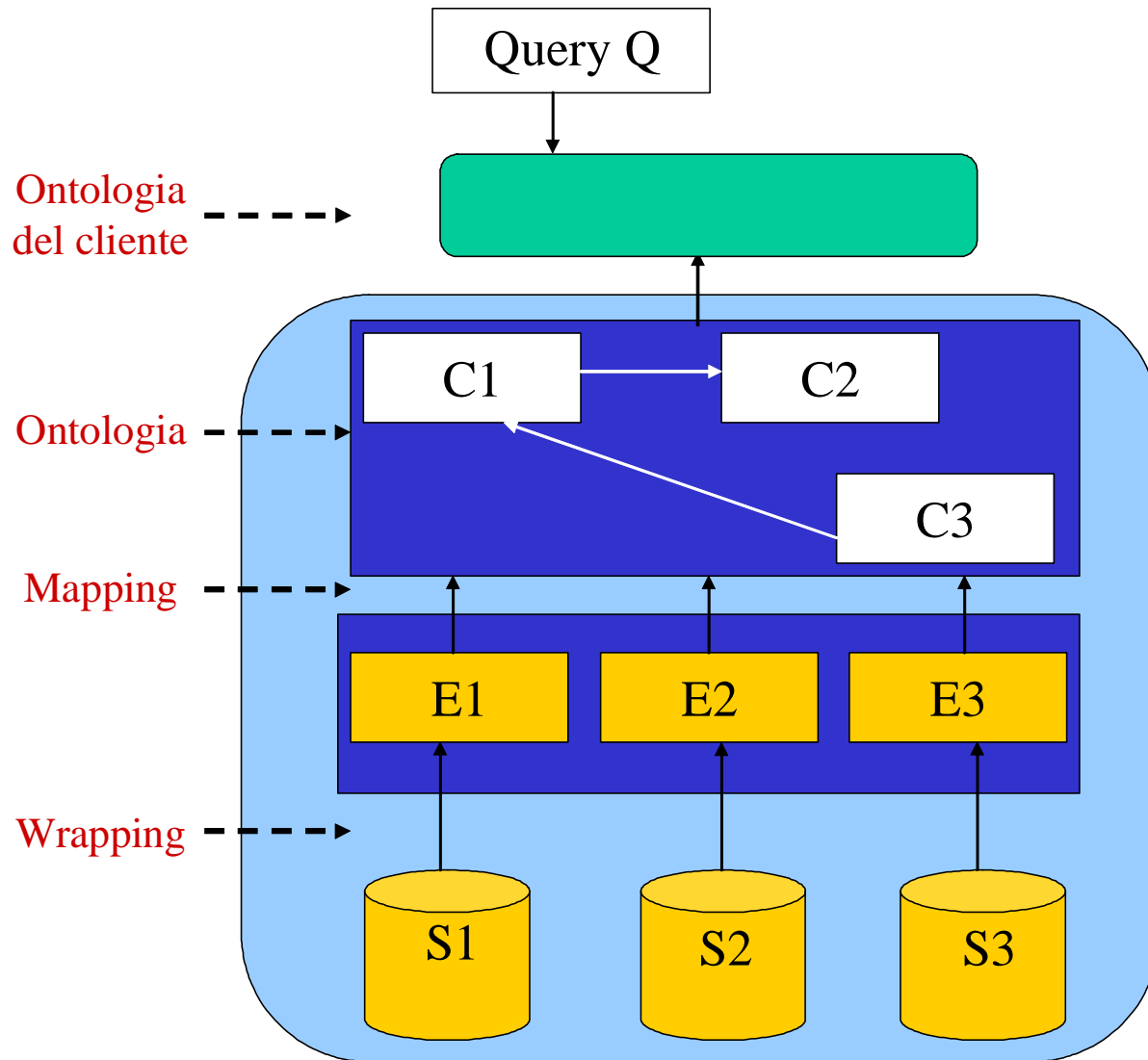
- Cosa fa

- consente di definire ontologie e le corrispondenze con sorgenti di dati
- supporta il progettista nell'analisi e nella gestione della specifica
- riformula query espresse sulle ontologie in query espresse sulle sorgenti

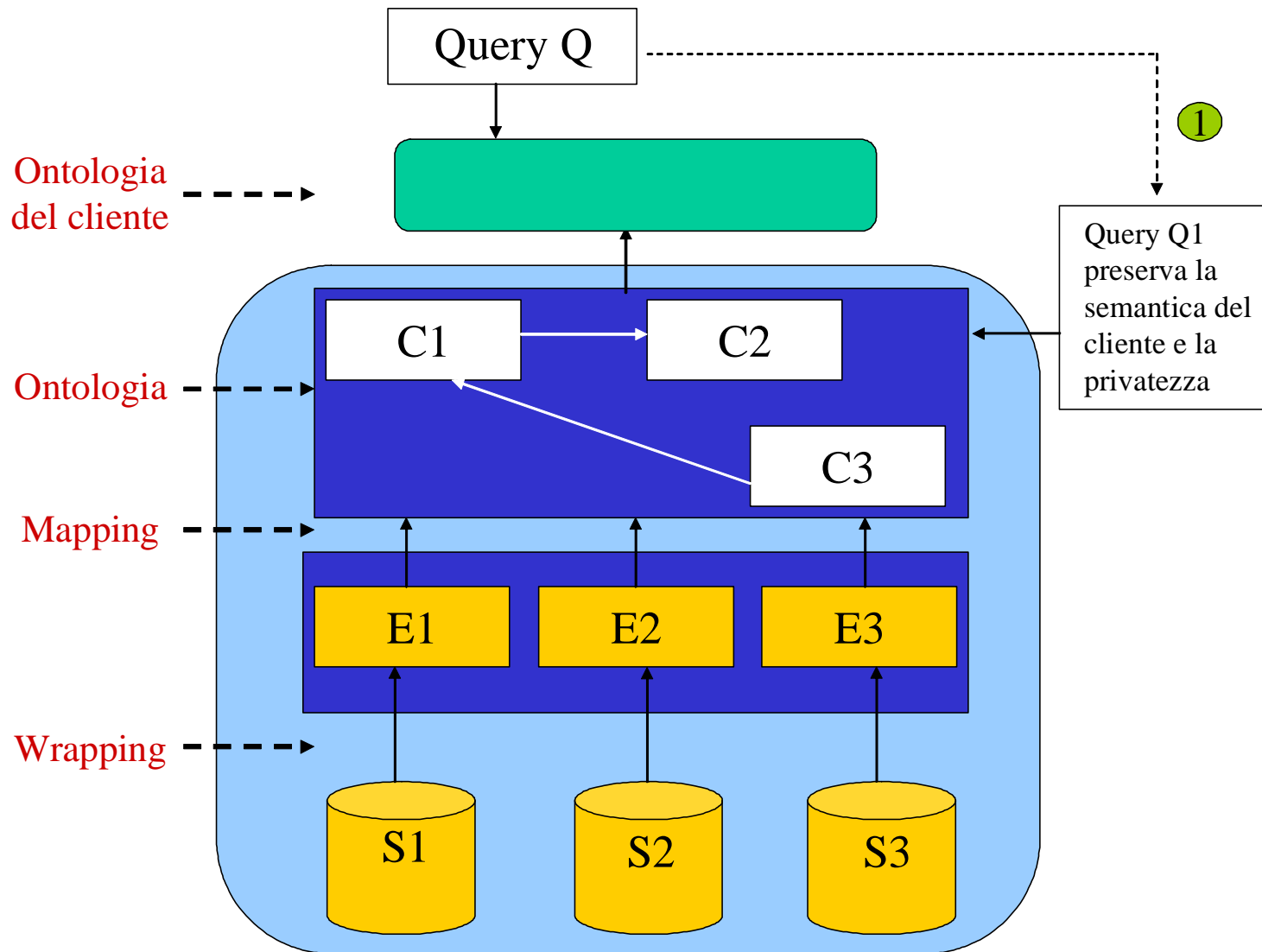
- Quali vantaggi offre

- ragiona logicamente in modo onniscente sulle ontologie, consentendo anche la validazione della specifica e la spiegazione delle inferenze
- deduce esattamente le query da spedire alle sorgenti (risolve il cosiddetto “what to ask problem”)
- ha lo stesso costo computazionale di un sistema di gestione di basi di dati, ed è quindi in grado di gestire sorgenti con enormi quantità di dati
- è scientificamente provato che **non ci può essere un sistema più espressivo con le stesse caratteristiche computazionali**

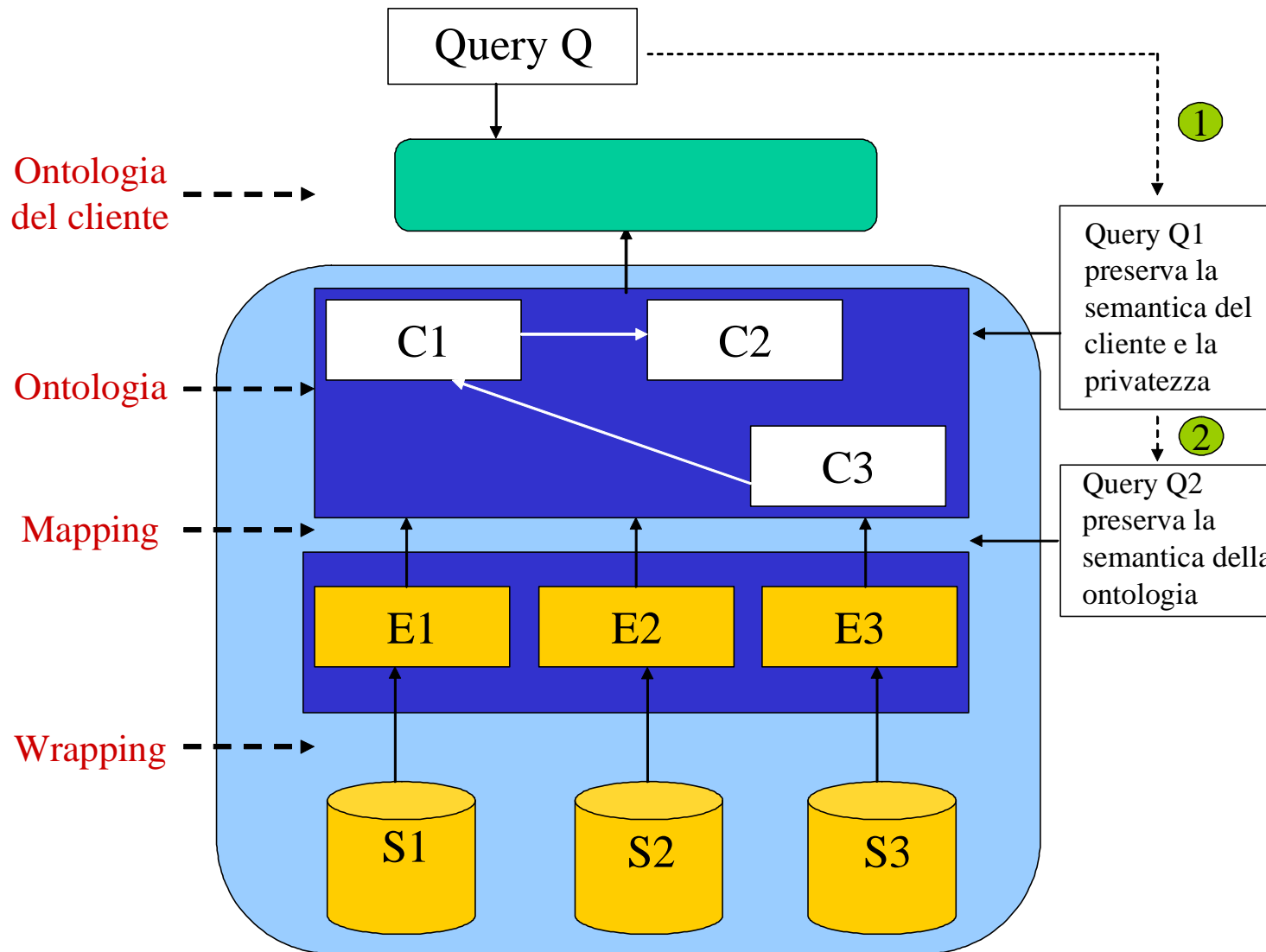
# Come QuOnto risponde alle query



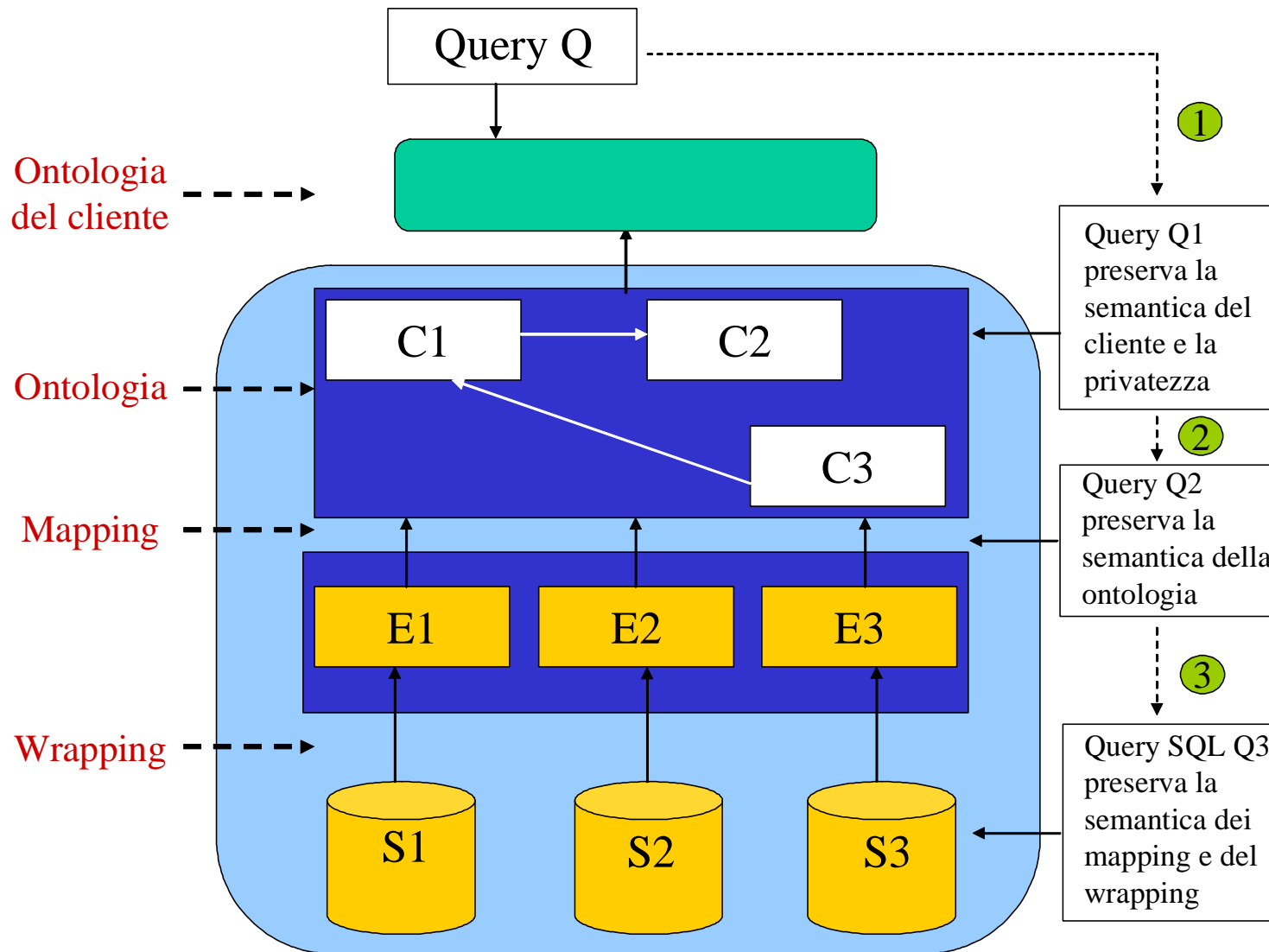
# QuOnto: fase 1 – riformulazione sulla ontologia del cliente



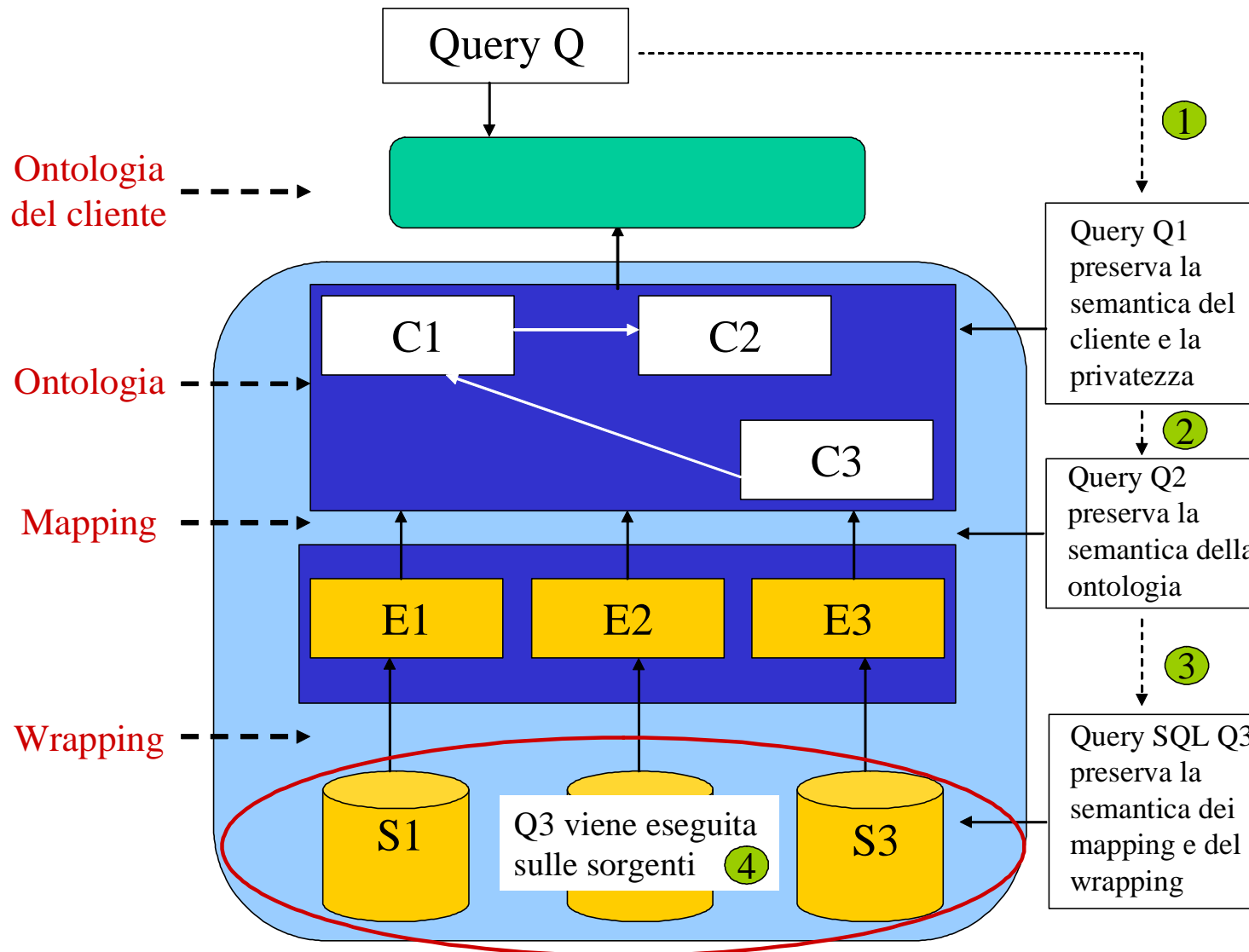
# QuOnto: fase 2 – riformulazione sulla ontologia



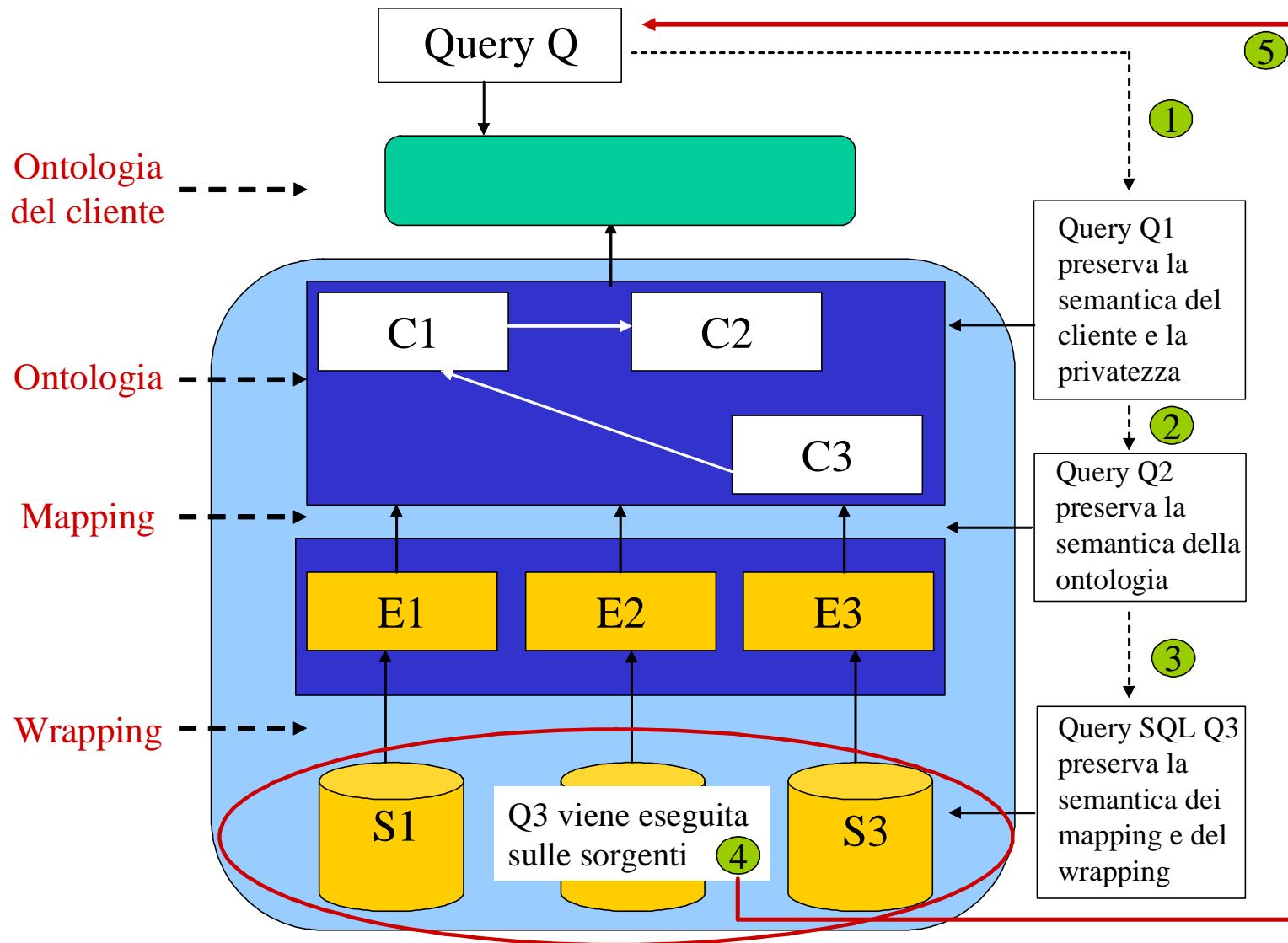
# QuOnto: fase 3 – riformulazione sulle sorgenti



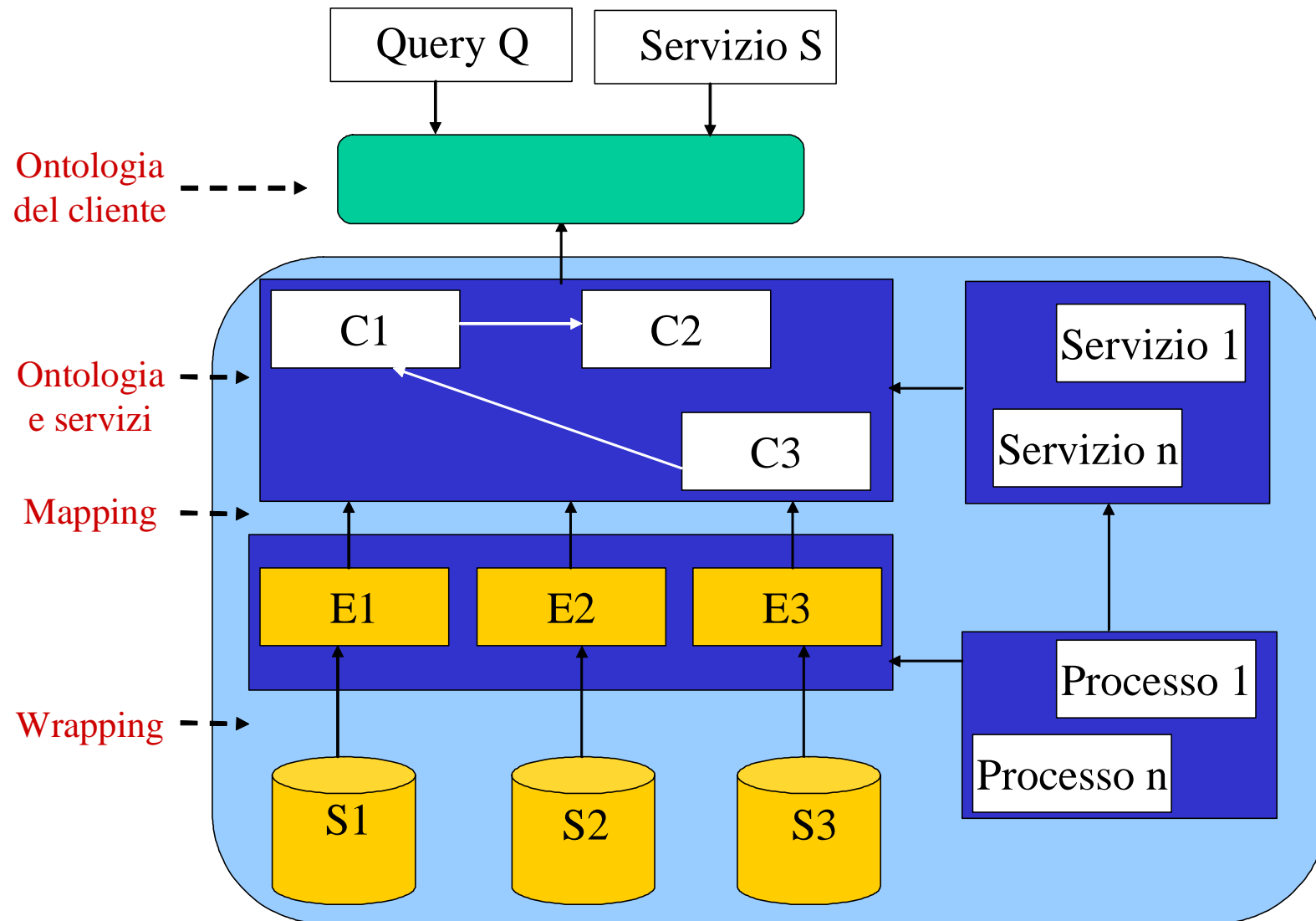
# QuOnto: fase 4 – valutazione query della SQL sulle sorgenti



# QuOnto: fase 5 – risposta alla query originaria



# Estensione di QuOnto ai servizi



## L'approccio Colombo per l'integrazione di dati e servizi

- **Colombo** è un sistema in via di sviluppo nell'ambito di un progetto di ricerca condotto da DIS-UNIROMA1 e Bell Lab. (Dr. Richard Hull), con lo scopo di realizzare un sistema che sia in grado di ragionare sui servizi
- Ogni servizio disponibile viene descritto a livello della ontologia, in termini di precondizioni, postcondizioni e specifica della dinamica (astrazione sul processo)
- Opportuni mapping creano le corrispondenze tra servizi e processi software
- Quando un cliente esprime una richiesta di servizio in termini di precondizioni e postcondizioni, il sistema tenta di sintetizzare automaticamente il servizio richiesto utilizzando i servizi disponibili (**composizione automatica**)
- Sebbene i tipi di ragionamento di Colombo siano più complessi di quelli di QuOnto, quest'ultimo viene invocato ogni volta che è necessario rispondere alle query sui dati e ragionare sulla ontologia

## Integrazione e cooperazione

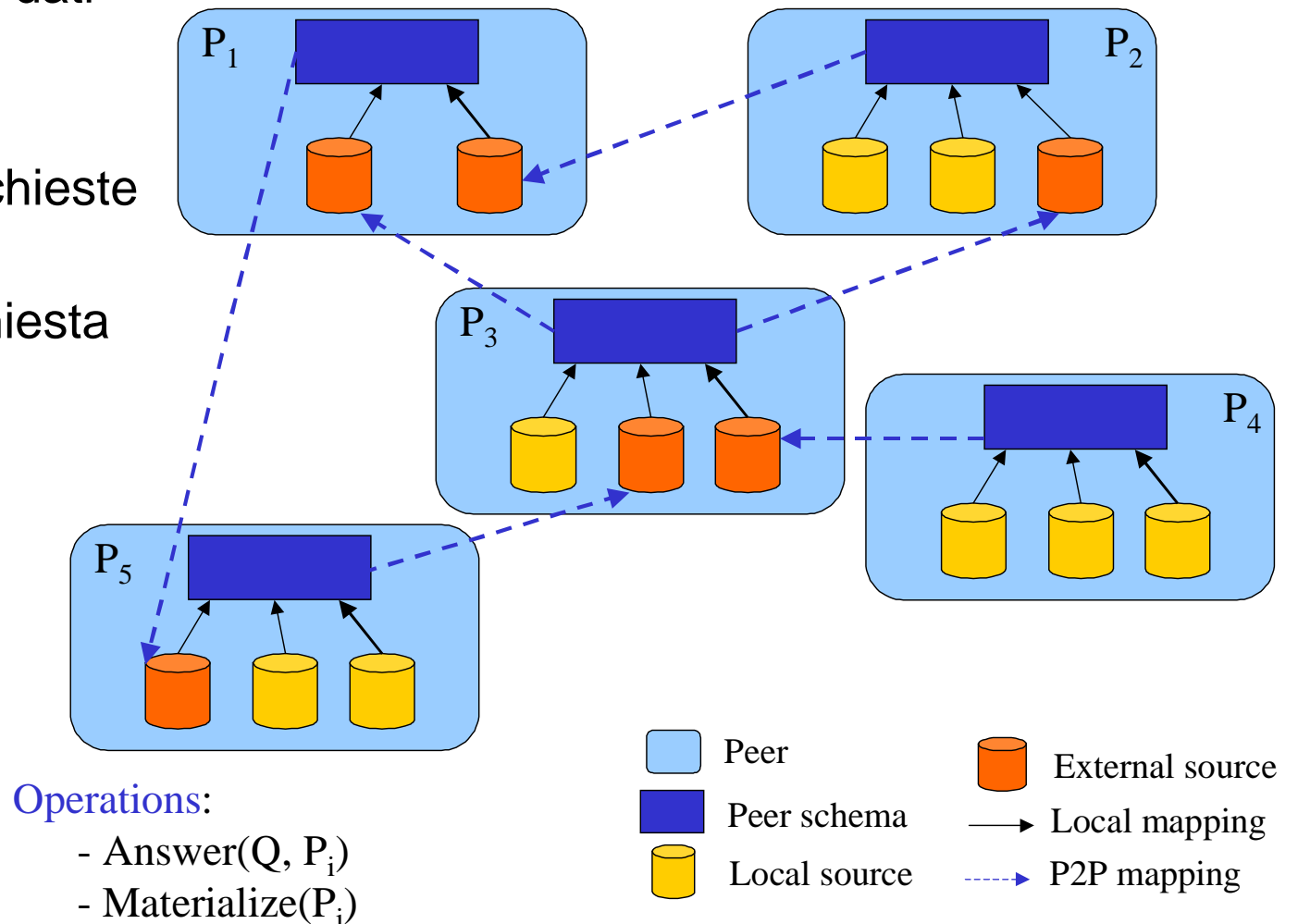
Avevamo distinto in precedenza tra:

- **Integrazione**: basata su mediatore
- **Cooperazione**: integrazione peer-to-peer di dati e servizi

rete di sistemi informativi autonomi (ciascuno visto come un sistema di integrazione basato su mediatore) collegati concettualmente tra loro, senza un unico schema globale

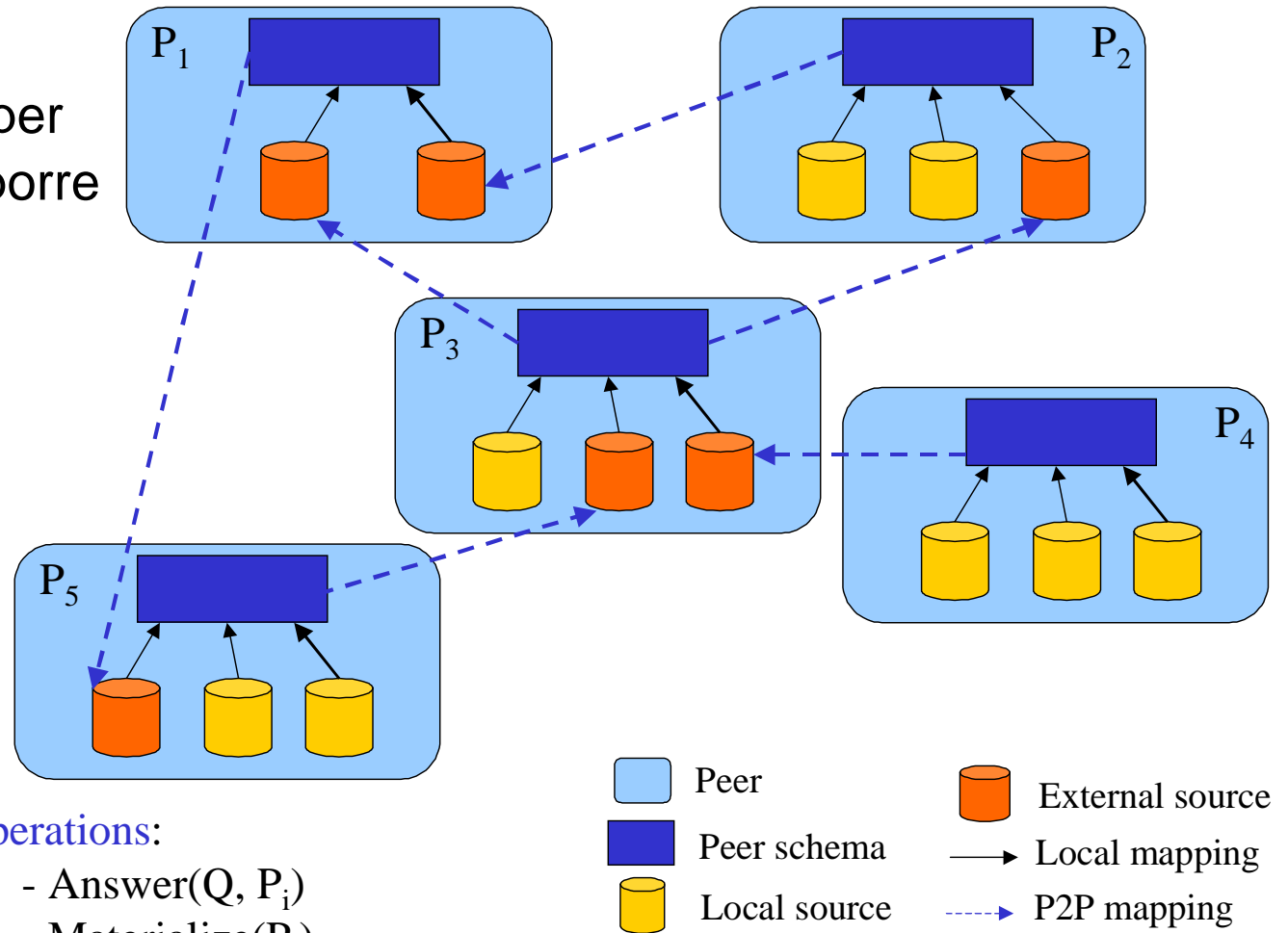
# Cooperazione: integrazione peer-to-peer di dati e servizi

- Diversi nodi (peers), **senza uno schema globale universale**
- Ogni nodo è un sistema di integrazione
- I mapping sono sia tra dati sia tra servizi
- Mapping dinamici
- Il cliente esprime le richieste ad un peer
- Il sistema serve la richiesta in modo distribuito



# Cooperazione mediante Colombo

- Ogni nodo è gestito da Colombo
- I mapping sono gestiti da Colombo
- Algoritmo distribuito che chiama il ragionatore di Colombo all'occorrenza per integrare dati e per comporre servizi



## Conclusioni

- Sistemi avanzati per gestire ontologie sono già disponibili, sulla base delle ricerche svolte negli ultimi vent'anni. La comunità scientifica italiana è stata ed è una protagonista assoluta di queste ricerche, anche se ha usufruito di finanziamenti nazionali irrisori, di **alcuni ordini di grandezza** inferiori rispetto al panorama internazionale
- **QuOnto** è il primo sistema per l'integrazione semantica di grandi quantità di dati; i suoi vantaggi e le sue caratteristiche sono scientificamente dimostrabili, ed è stato sperimentato in diversi contesti, inclusi due progetti europei (SEWASIE – coordinatore Prof.ssa Sonia Bergamaschi, e INFOMIX – coordinatore Prof. Nicola Leone)
- La versione di QuOnto per la cooperazione a livello di dati (**Hyper**: integrazione peer-to-peer dei dati) è stata già definita e sviluppata, anche con il supporto fondamentale di due **IBM Shared University Research** dato al DIS-UNIROMA1

## Conclusioni

- **Colombo** è un tentativo ancora in fase di sviluppo per aggiungere a QuOnto la capacità di integrare servizi
- L'integrazione è un problema complesso, e molti aspetti importanti (come scoprire automaticamente i mapping, come gestire eventuali inconsistenze tra sorgenti, ecc.) sono stati tralasciati in questa presentazione
- L'adozione di metodologie e tecnologie basate sulla semantica **non** può considerarsi un miracoloso toccasana. Il lavoro da parte del progettista è ancora molto rilevante (cfr. **A. Einstein: Things should be made as simple as possible, but not any simpler.**)
- Tuttavia, sistemi come QuOnto, Hyper, Colombo possono fare un decisivo salto di qualità sia nella fase di specifica e progetto del sistema di integrazione, sia nella fase di esercizio dello stesso
- La prospettiva dell'integrazione semantica dei servizi è molto stimolante e, ancora grazie alla possibilità di ragionare sulle ontologie, realistica

# Ringraziamenti

I principali compagni di viaggio delle ricerche su Quonto e Colombo:

- Diego Calvanese
- Giuseppe De Giacomo
- Richard Hull
- Domenico Lembo
- Massimo Mecella
- Riccardo Rosati
- Moshe Y. Vardi