

UNIVERSITÀ DEGLI STUDI DI UDINE

DIPARTIMENTO DI SCIENZE MATEMATICHE, INFORMATICHE E FISICHE

CORSO DI LAUREA IN TECNOLOGIE WEB E MULTIMEDIALI

TESI DI LAUREA

CANDIDATO

Andrea Simeone

RELATORE

Prof. Marino Miculan

Anno accademico 2017-2018

CONTATTI DELL'ISTITUTO







---

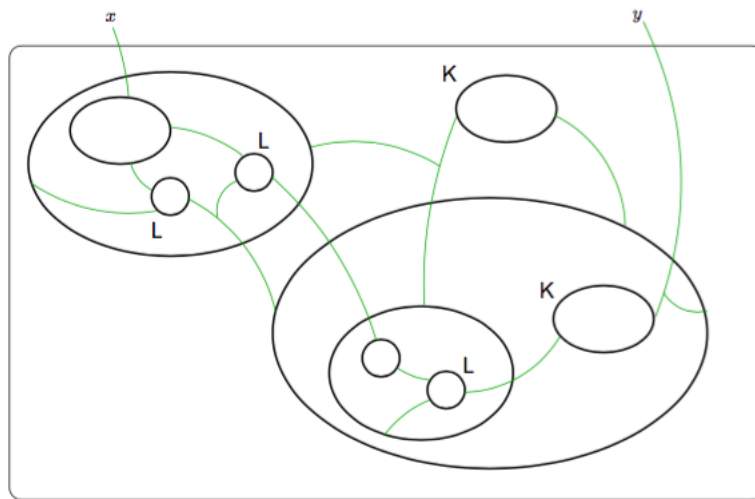




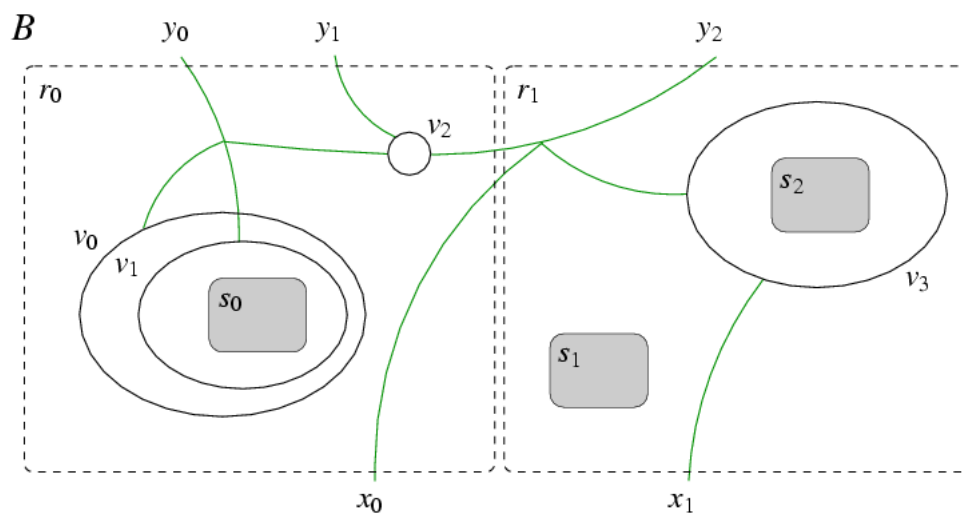


---

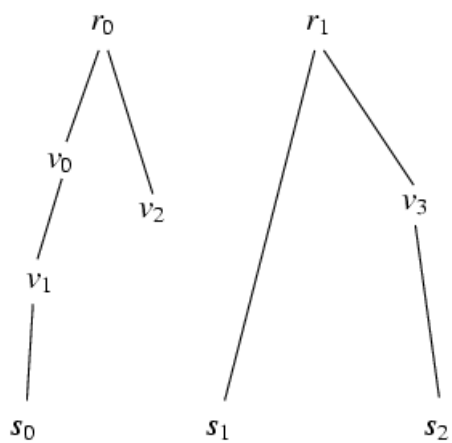
<sup>1</sup>Anche il Laboratorio di Sistemi Distribuiti dell'Università di Udine, all'interno di una libreria di implementazione di bigrafi e BRS, ha sviluppato un sistema di visualizzazione di bigrafi. Per maggiori informazioni: <http://mads.uniud.it/wordpress/downloads/libbig/>







(a) Bigrafo

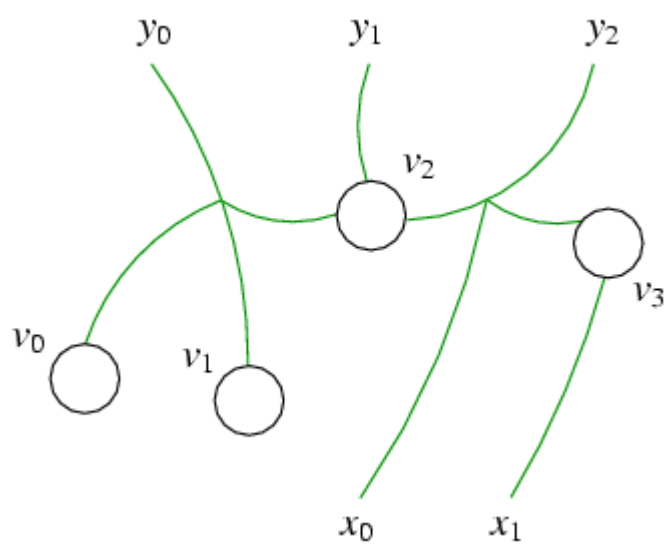


(b) Place graph

---

$o \quad 1 \quad 2$

$o \quad 1$



$0 \quad 1$

$o \quad 1 \quad 2$

$P \quad P \quad L \quad L$

$P \quad P$

$P$

$P$

$L \quad L$

$L$

$L$

$0 \quad 1$

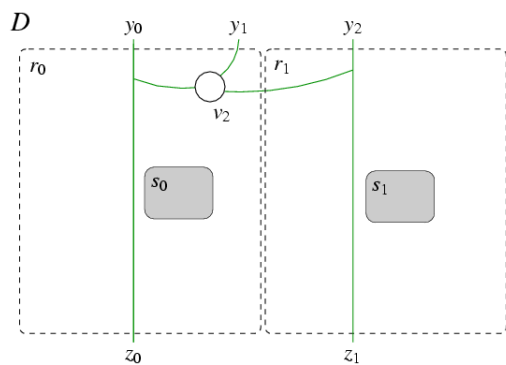
$o \quad 1 \quad 2$

$0 \quad 1$

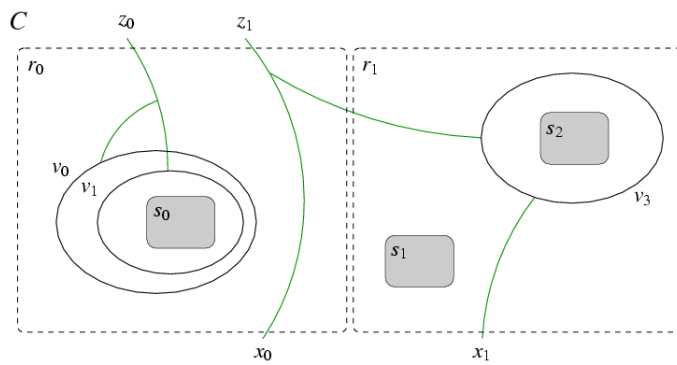
$o \quad 1$

$0 \quad 1$

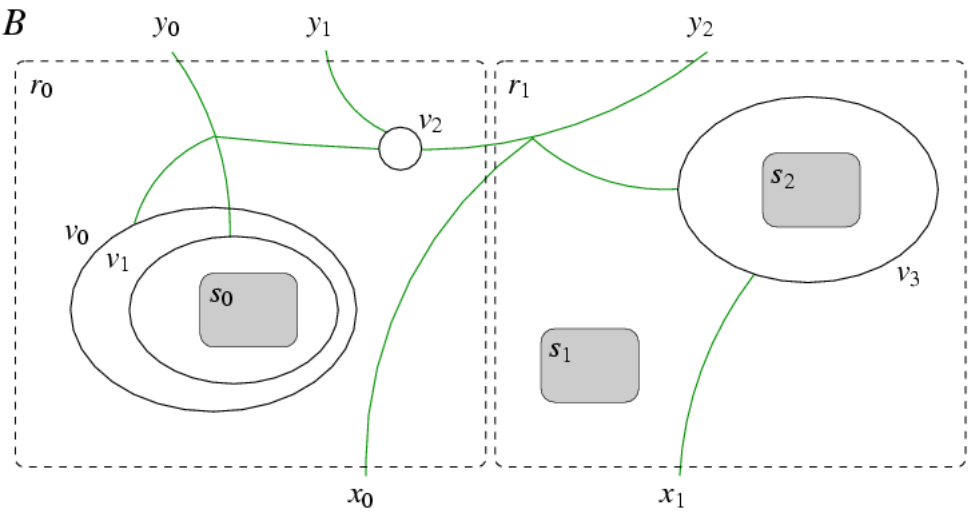
$o \quad 1 \quad 2$



(a) Bigrafo  $D$



(b) Bigrafo  $C$



$i \quad i \quad i \quad i \quad i$

$i \quad i$

$F \quad F \quad F \quad F \quad G \quad G \quad G \quad G$

$F \quad G \quad F \quad G \quad F \quad G \quad F \quad G$

$F \quad G \quad F \quad G \quad F \quad G \quad F \quad G \quad F \quad G$

$G \quad F \quad F \quad G$







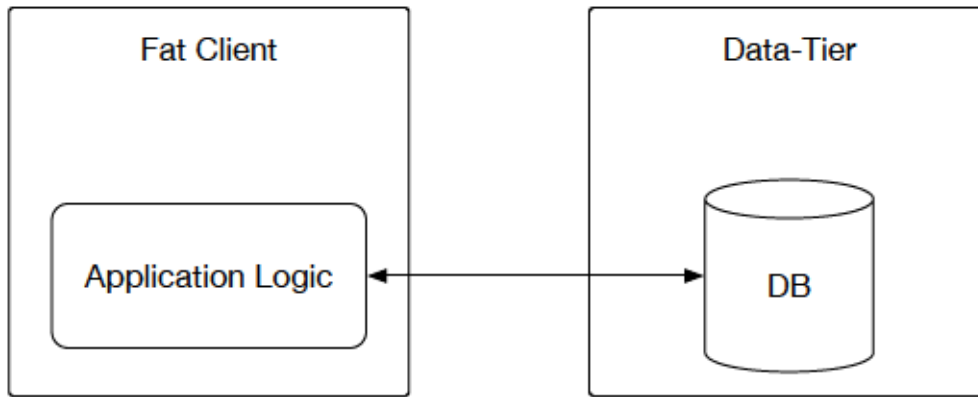


Figura 3.1: Esempio di sistema Two-Tier

## 3.2 Progetto: architettura di riferimento

La definizione dei requisiti dell'utente ha naturalmente influito nella definizione dei requisiti del sistema indirizzando innanzitutto le scelte architetturali del sistema, successivamente la scelta degli strumenti software più adatti alla realizzazione dello stesso. Fino dalle fasi iniziali della stesura dei requisiti software è parso naturale scegliere come architettura di riferimento per il nostro sistema quella **Two-Tier** (figura 3.1) (*"a 2 strati"* comunemente chiamata Client-Server) con **Fat Client** ovvero quella tipologia di client che contiene oltre alla visualizzazione e interazione con l'utente anche la parte relativa all'applicazione. Il modello *Two -Tier* è un pattern architetturale che prevede, per l'appunto, la suddivisione in 2 strati (detti *layer*) della nostra applicazione in cui ogni strato si occupa di un preciso aspetto all'interno dell'applicazione. I layer sono:

1. il **Presentation layer & Application processing** ovvero lo strato che si occupa di fornire le funzionalità richieste all'applicazione, della presentazione dei risultati della computazione all'utilizzatore del sistema e della interazione dell'utente con essa collezionando gli input dell'utente;
2. il **Data management layer** ovvero lo strato che gestisce i dati persistenti della nostra applicazione.

### 3.2.1 Descrizione dei moduli

Si procede ora nella descrizione dettagliata di ogni singolo layer.

#### 3.2.1.1 Presentation & Application Layer

È il layer fondamentale della nostra applicazione, in quanto si occupa delle operazioni fondamentali del nostro programma:

- richiesta e ricezione via *HTTP* (HyperText Transfer Protocol) del file contenente la struttura dati in cui è descritto il bigrafo;
- parsing del file ricevuto e estrapolazione dei dati;

---

---

---

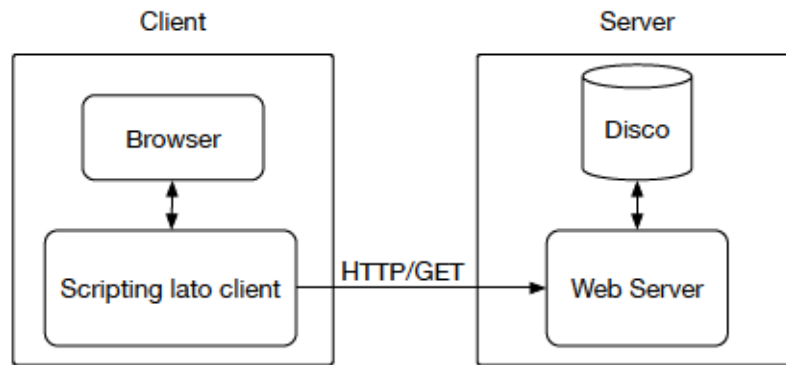


Figura 3.2: Schema di richiesta risorse

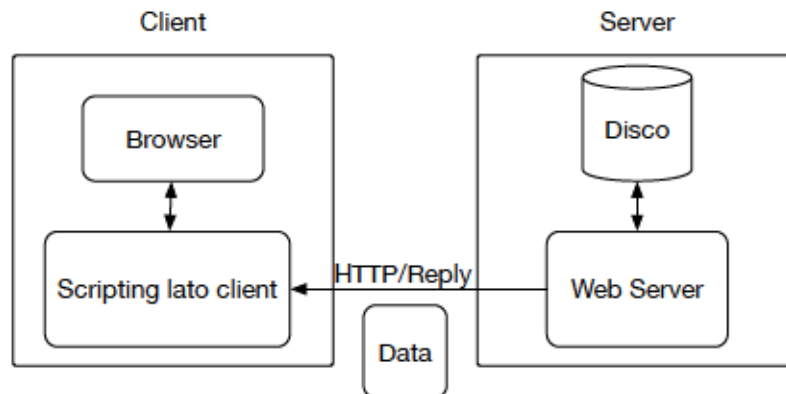


Figura 3.3: Schema di ricezione risorse

### 3.2.2.2 Risposta del server

Il server ricevuta la richiesta invio del file richiesto dal client provvederà alla sua ricerca su disco e al suo invio mediante HTTP. Il client alla ricezione dei dati, codificati in *JSON* o *XML*, eseguirà la funzione di callback con le relative operazioni di estrapolazione dei dati. Il client dovrà successivamente provvedere all'estrapolazione dei dati e alla creazione della pagina *HTML* contenente la visualizzazione del bigrafo. La figura 3.3 illustra schematicamente il processo di ricezione dei dati.



---

1

2

---

<sup>1</sup>ad esempio: <http://www.nytimes.com/interactive/2012/02/13/us/politics/2013-budget-proposal-graphic.html>

<sup>2</sup>ad esempio <https://www.theguardian.com/world/interactive/2013/feb/12/state-of-the-union-reading-level>

---

<sup>3</sup>La prima Recommendation del W3C di SVG 1.0 è visibile al link <https://www.w3.org/TR/2001/REC-SVG-20010904/>

<sup>4</sup>La Recommendation del W3C di SVG 1.1 è visibile al link <https://www.w3.org/TR/SVG11/>

<sup>5</sup>Attualmente è visibile una Candidate Recommendation di SVG 2.0 al link <https://www.w3.org/TR/SVG2/>

---



---

---

---

---

---

---

---

---

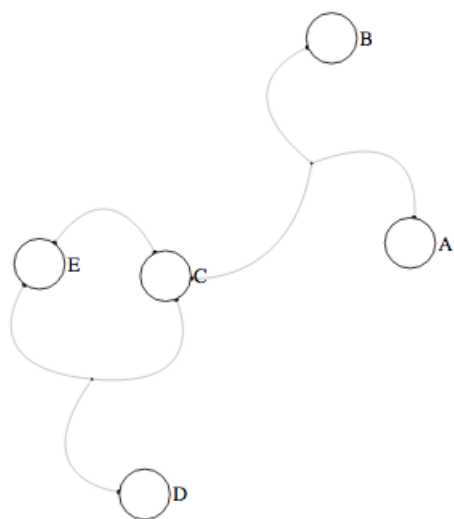
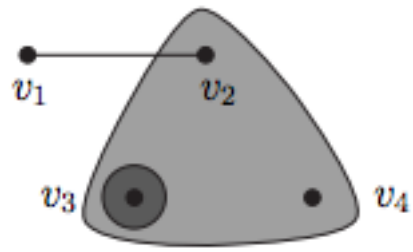


---

1 2 3 4                      1 2        2 3 4        3

---

<sup>6</sup>Per scaricare il pacchetto visitare la pagina <https://github.com/AndreaSimeone/d3-hypergraph> oppure cercare *D3-hypergraph* tramite **npm** (node package manager)



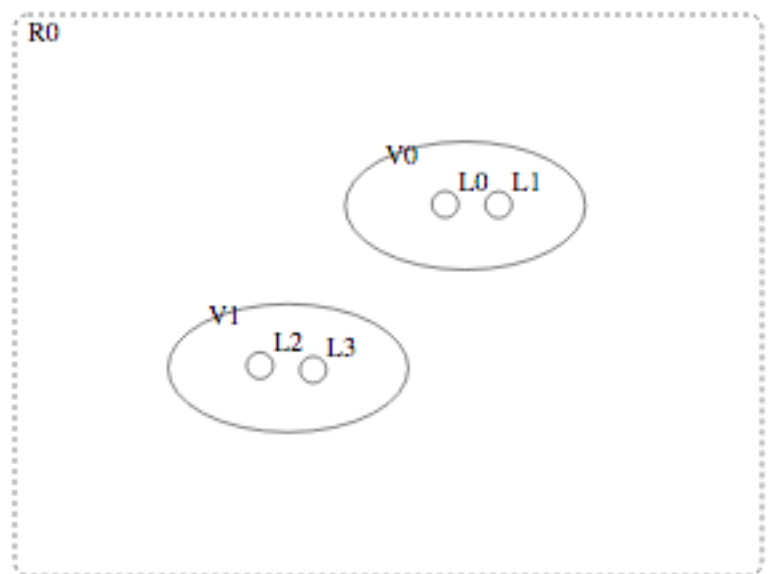
---

---

---

---

(a) file Json che descrive la struttura del bigrafo



(b) Esempio di inclusione tra nodi ottenuta con 4 layout Force

---

---

---

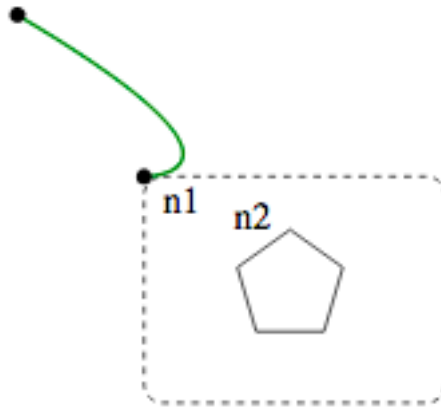
---

---

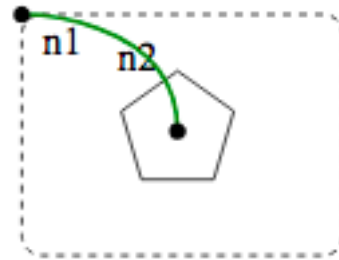
---

---





(a) Link senza trasformazione del nodo padre

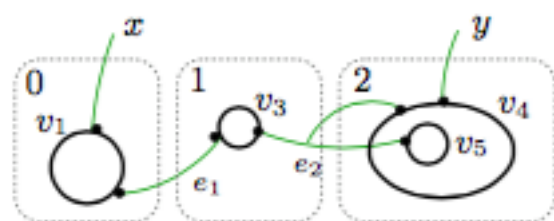


(b) Link con trasformazione del  
nodo padre

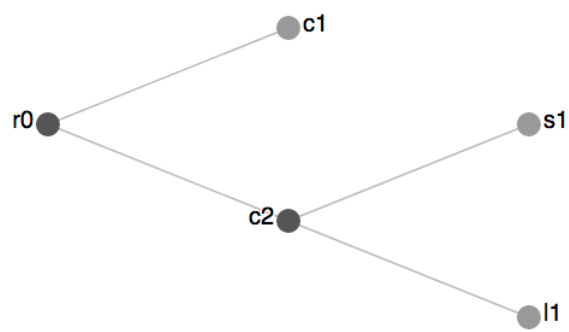
---

<sup>7</sup>Per maggiori informazioni: <http://www.cytoscape.org/>

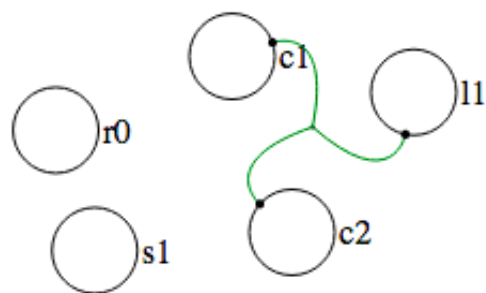
<sup>8</sup>libreria realizzata da Kevin Lindsey e scaricabile all'indirizzo <https://github.com/thelonious/kld-intersections>



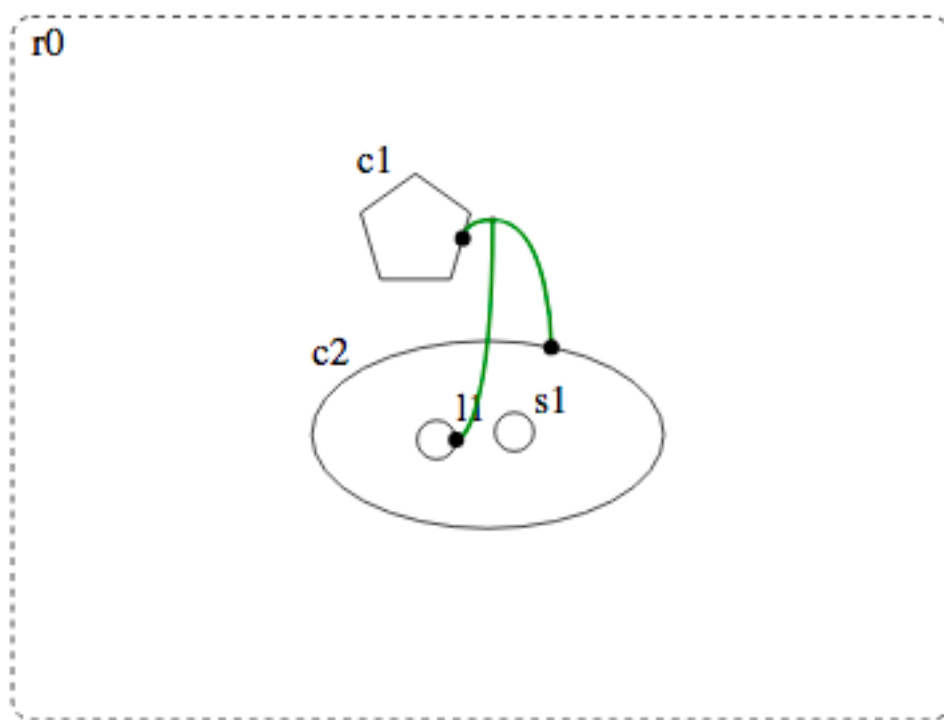




(a) Place Graph



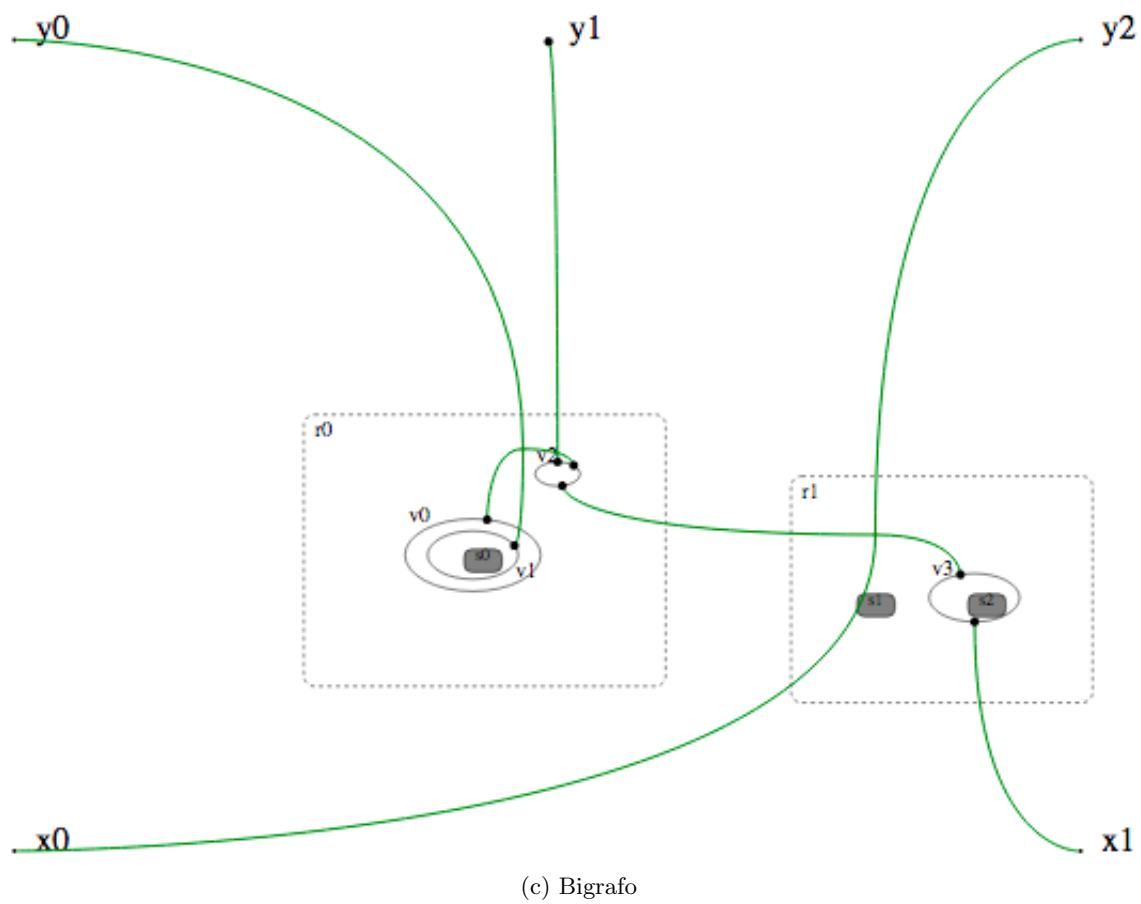
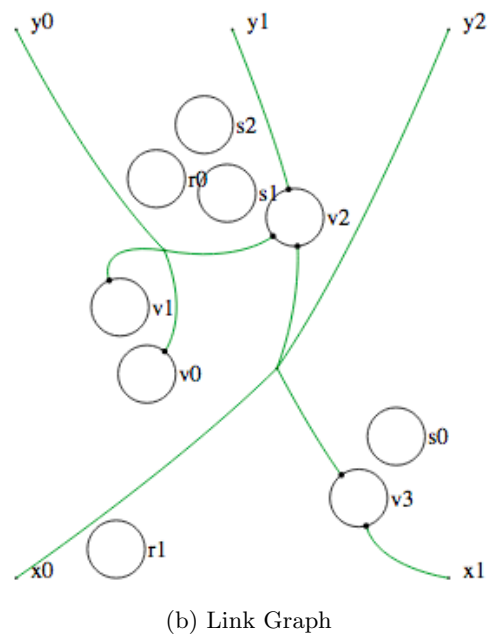
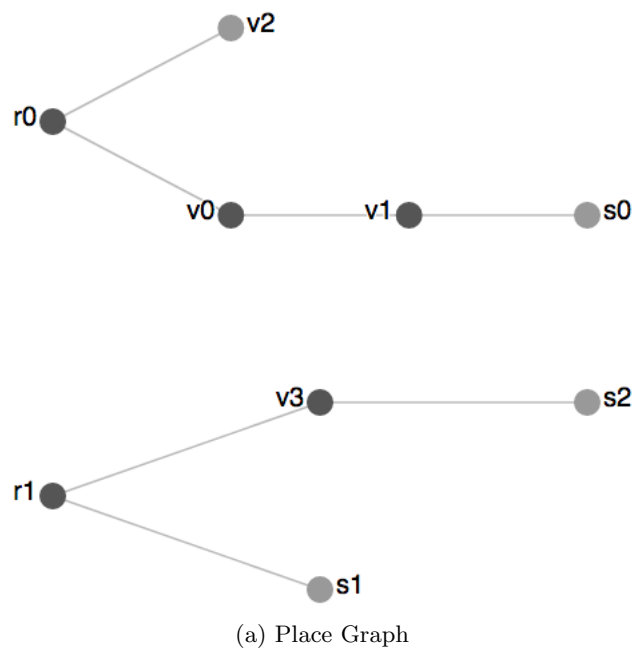
(b) Link Graph



(c) Bigrafo

---

---



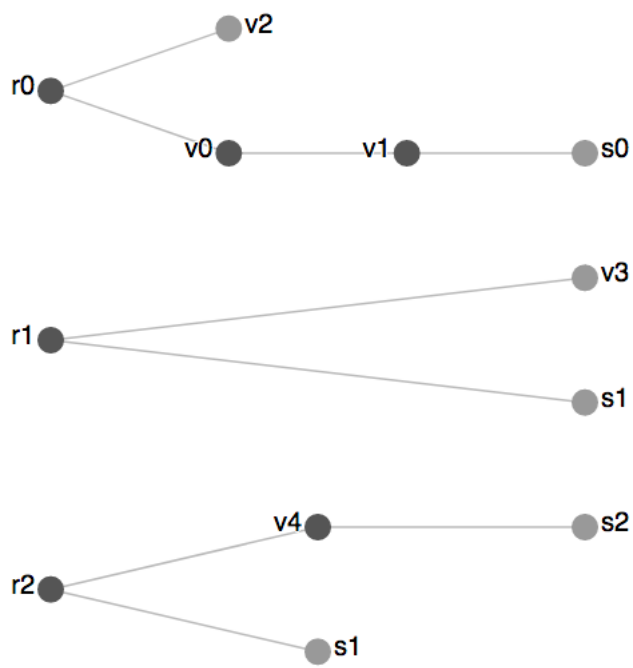
---

---

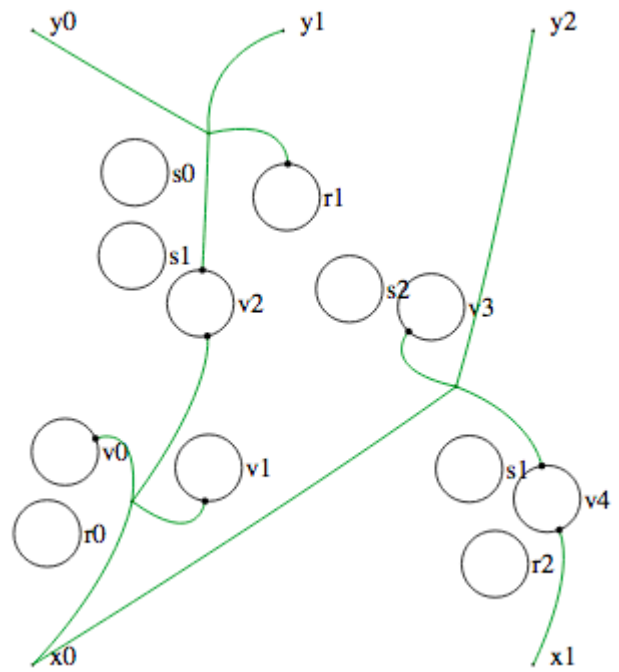


---

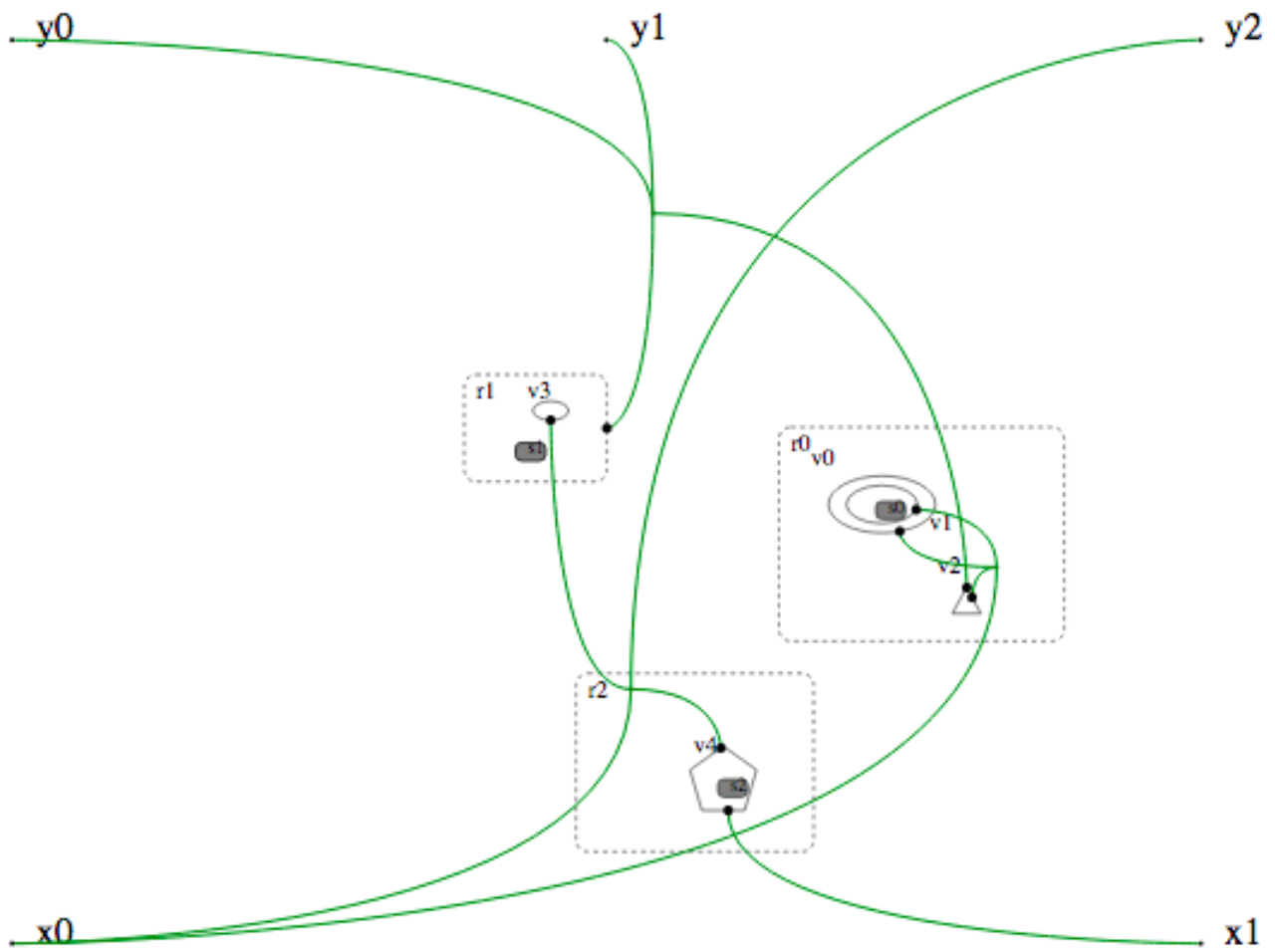
---



(a) Place Graph



(b) Link Graph



(c) Bigrafo



---





