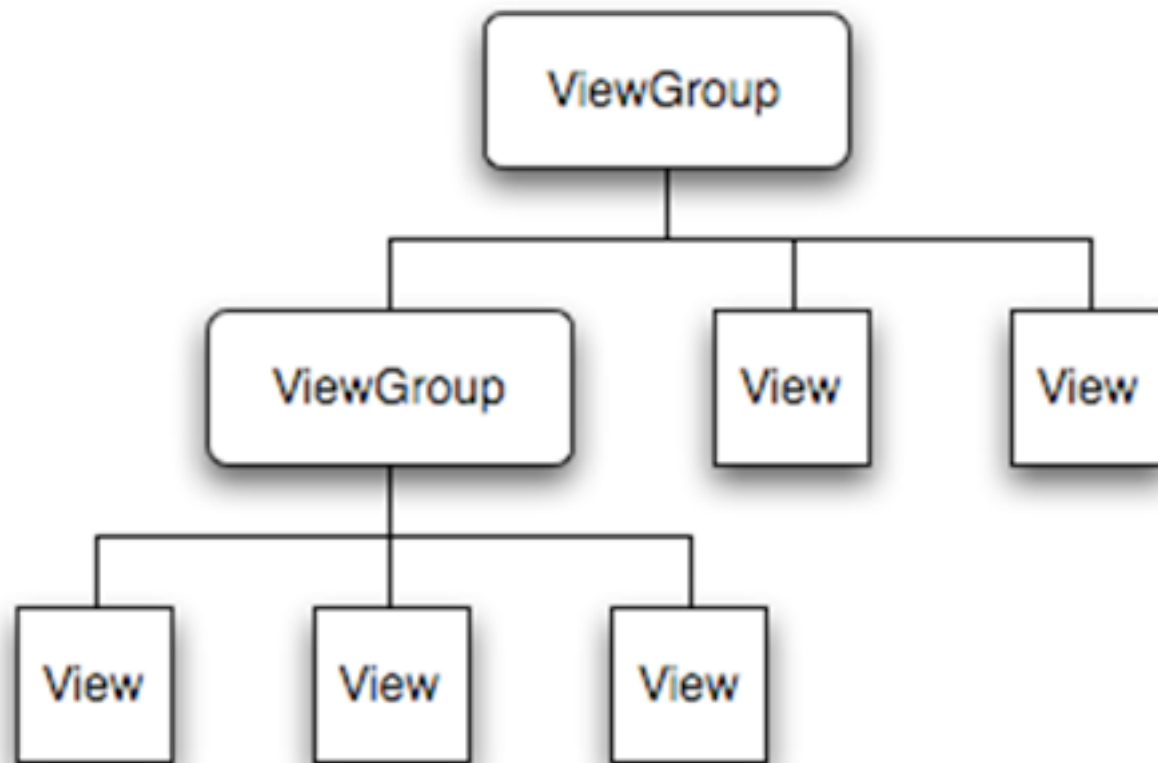


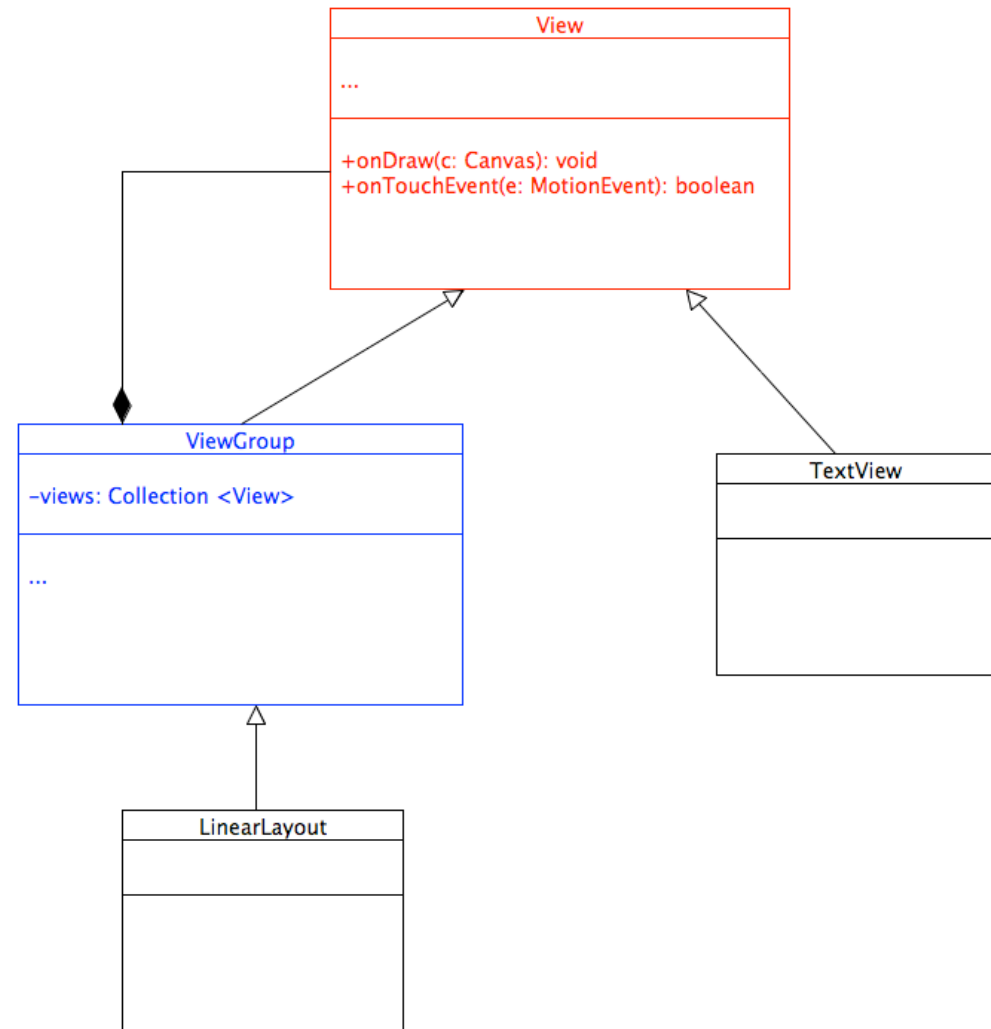
# Elementi grafici

# Gerarchia componenti grafici

- I componenti grafici formano un albero



# View, ViewGroup e Layout



- Composite Pattern
- Single Thread Model

# Albero delle View

- Il composite pattern fa sì che dalla root view si venga a generare un'albero di view
  - L'attraversamento dell'albero cambia l'ordine di visualizzazione
- Android garantisce che
  - i nodi sono attraversati partendo dal root
  - i nodi più vicini al root sono attraversati prima
  - i nodi di pari livello sono attraversati secondo l'ordine di definizione

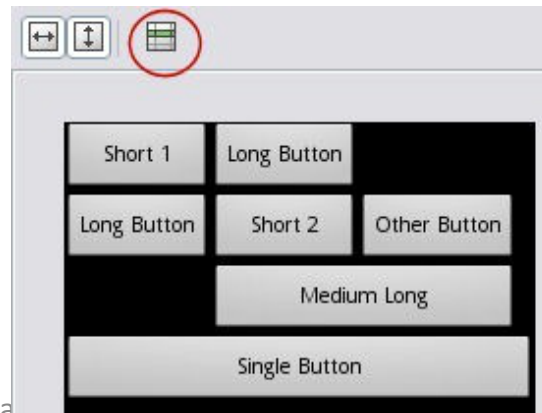
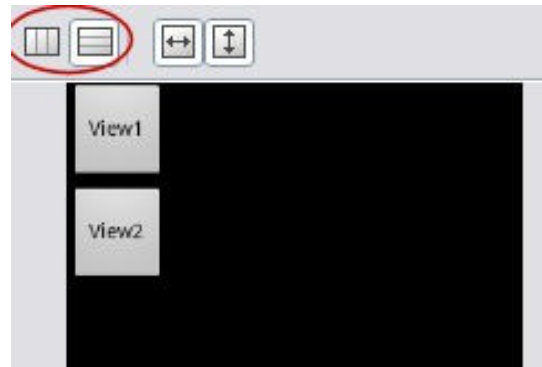
# XML e View

- Le view possono essere create
  - in java
  - in xml e poi convertite in oggetti java
- L'operazione di conversione si chiama **inflating**
  - da un documento xml genera una view
- Per ottenere il riferimento all'oggetto uso
  - `public View findViewById (int id)`
- Inflating del root ne metodo onCreate
  - `public void setContentView(int id)`

# Layout

- Un layout è
  - una specializzazione di ViewGroup
  - posiziona le View contenute in un modo “specifico”

- Layout predefiniti
  - LinearLayout
  - RelativeLayout
  - TableLayout
  - FrameLayout



# Linear Layout

- Allinea le view contenute
- Si specifica il verso di allineamento
  - `android:layout_orientation`
  - `public void setOrientation (int orientation)`
- Si può specificare la dimensione relativa
  - `android:layout_weight`
- nell'altra direzione è possibile gestire l'allineamento
  - `android:layout_gravity`
- Progetto `LinearLayoutTest`

