Some notes on functionalization of mashup composition

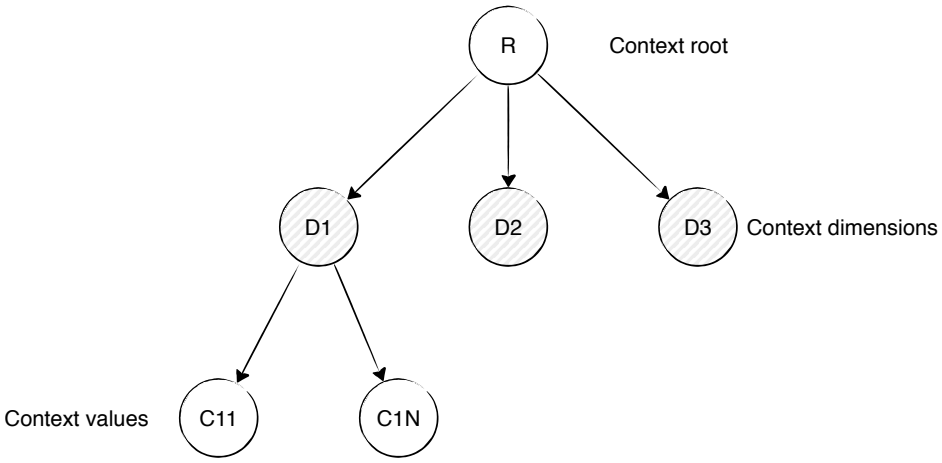
Vittorio Zaccaria

September 22, 2016

Contents

1 Background

It is well understood that the CDT model allows for a declarative expression of the type of a context c which we will broadly assume to be values/parameters associated with the situational needs in which the query must be answered.



Each dimension D_d has one value of type indicated by one of children $C_{d,i}$, also called **conceptual nodes**. Each children C provides a **view**, i.e. some way of ex-

tracting interesting data from a particular data store (being it a database or a online service). Effectively, this is a function from the context c to a set of actual values of the store. We might see this as the information value stored into a standard tree leaf.

```haskell
data Context       = (InterestTopic, Role, Maybe Location, Maybe Interface)
data InterestTopic = Orders (DateRange) | Clients | Food FoodInfo
data View a        = Context -> a

cdt :: Tree (View a)
```

The actual view creation can be seen as a `fold` operation ϕ across the tree, where the values effectively folded are the views. The fold operation might depend on the context values as well

$$\phi(c, v_a, v_1) \to v_a$$