

Making Semantic Bidirectionalization More Applicable

Meng Wang (Chalmers Univ. of Tech.)

in collaboration with Kazutaka Matsuda and Shayan Najd

Banff-BX, 2-6 Dec 2013

Existing PL Approaches

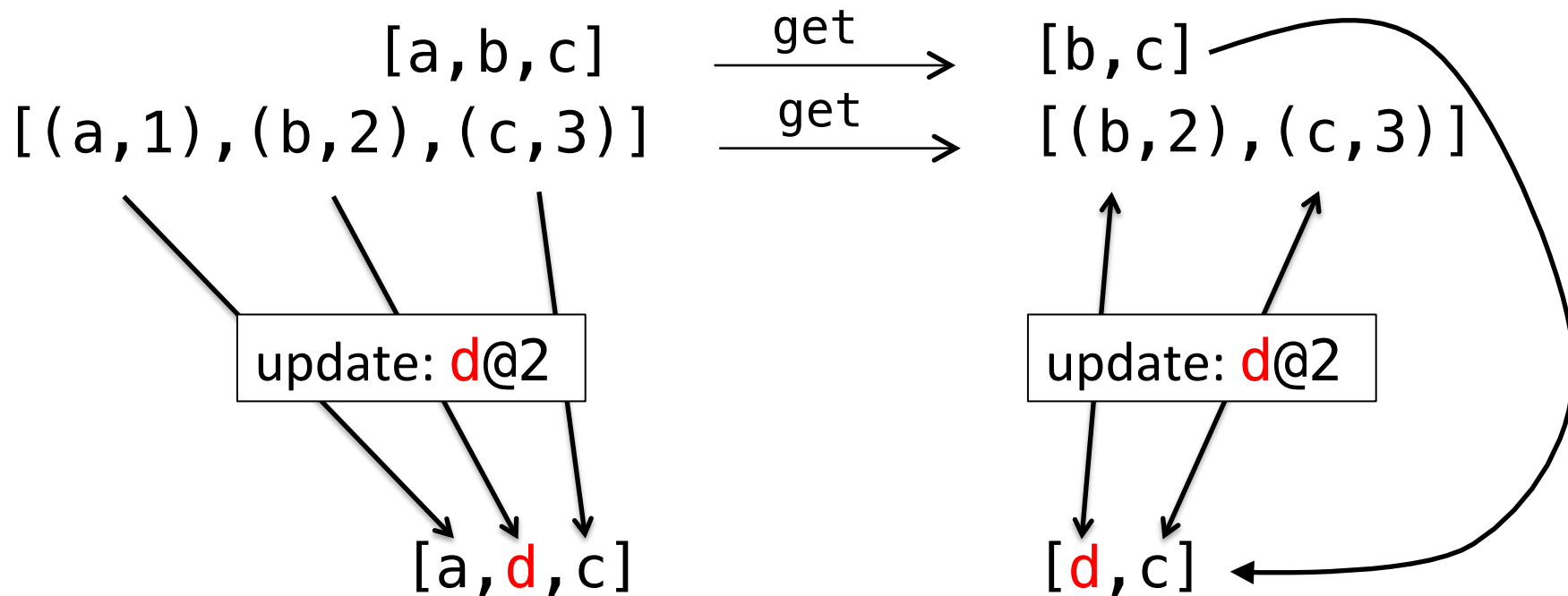
- **Combinator languages**
 - Lenses [Foster et al. TOPLAS'07, ...]
 - ...
- **Bidirectionalization**
 - Syntactic [Matsuda et. at. ICFP'07]
 - **Semantic** [Voigtländer POPL'09]

Semantic Bidirectionalization

- Deriving `put` from `get` of type
$$\forall a. T_1 a \rightarrow T_2 a$$
without inspecting the definition
- Advantages
 - simplicity
 - expressiveness of `get`
- Correctness
 - free theorems [Wadler FPCA'89]

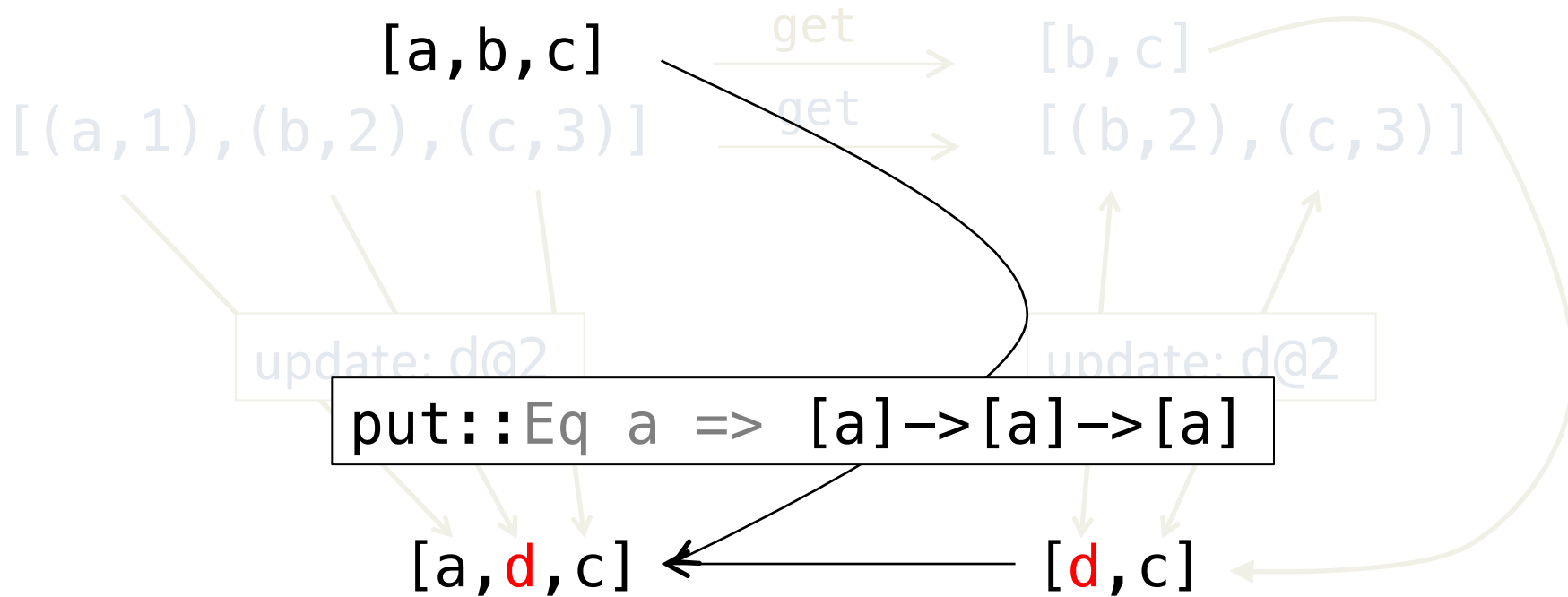
Constructing Put

Example: $\text{get} :: [a] \rightarrow [a] = \text{tail}$



Constructing Put

Example: $\text{get} :: [a] \rightarrow [a] = \text{tail}$

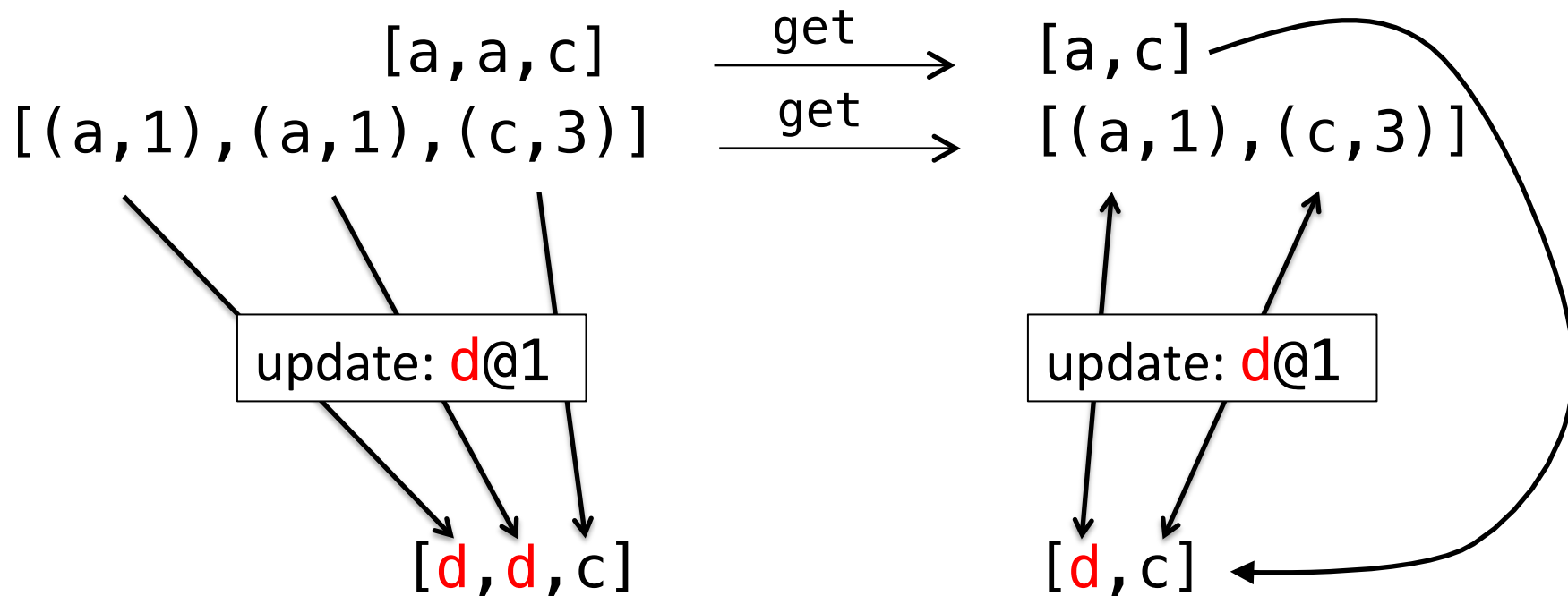


Extension

- Deriving put from get of type
 $\forall a. \text{Eq } a \Rightarrow T_1 a \rightarrow T_2 a$
without inspecting the definition

Constructing Put

Example: $\text{get} :: \text{Eq } a \Rightarrow [a] \rightarrow [a] = \text{nub}$



Summary

- Reviewed: [Voigtländer POPL'09]
 - Simplicity
 - Expressiveness of get
- Result:

Deriving put from get of type

$$\forall a. (\text{Eq } a, \text{Ord } a) \Rightarrow T_1 a \rightarrow T_2 a$$

without inspecting the definition

Limitations

- **Not** easy to extend to other polymorphic functions
- **Not** possible to deal with monomorphic functions

Advances

- ~~Not~~ easy to extend to other polymorphic functions
 - A more general implementation that handles all cases uniformly (PEPM'14)

```
filter :: (a -> Bool) -> [a] -> [a]
takeWhile :: (a -> Bool) -> [a] -> [a]
...
```

Advances

- ~~Not~~ possible to deal with monomorphic functions
 - A system with run-time recording of observations (PPDP'13) <https://bitbucket.org/kztk/cheap-b18n>

```
countWords :: String -> [(Int,String)]
countWords "hello clouds hello sky" =
    [(2,"hello"),(1,"clouds"),(1,"sky")]
```

XML Example

- get is monomorphic

```
<bib>
{
  for $b in doc("http://bstore1.example.com/bib.xml")/bib/book
  where $b/publisher = "Addison-Wesley" and $b/@year > 1991
  return
    <book year="{ $b/@year }">
      { $b/title }
    </book>
}
</bib>
```

from XML Query Use Cases

<http://www.w3.org/TR/xquery-use-case/#xmp-queries-results-q1>

Conclusion

- Semantic bidirectionalization scales to
 - most polymorphic functions
 - many monomorphic functions (through some code instrumentation)
- Structure change is still limited