

計算モデル: 2flow

by Haochen et al.



@haochenxie

自己紹介



にゃーです

Haochen Xie
@haochenxie

ぷるぐらみんぐ ちょっと できる/i could
program a little bit/我會一點點程式設計/je
peux programmer un petit peu

📍 Metaland 🔗 haochenxie.name
📅 Born March 19, 1995
📅 Joined March 2013

- Haochenです
- 理論よりプログラマーです
- 形式手法プロモーターです
- 猫、DSL、関数型プログラミング
大好きです
- コンサルティングやったり
会社やったり(閉じたり)

今日はDSLの話をしてします

今日はDSLの話をしません

DSLのために作った**計算モデル**のお話をします

計算モデルとは

- チューリングマシンやλ計算のような「計算」という概念を形式化する数学構造
- 一般的なコンピュータプログラムではRASP (= Random Access Stored Program)という計算モデルに基いている
- ※一般的=CやらRustやら; von Neumann architectureと呼ばれたりするものの中心部にも当たる
- 異なる計算モデルでは計算力(=計算できる問題)に差が無い(チャーチ=チューリングのテーゼ)ことが多いが、「計算」という概念の捉え方が異なる

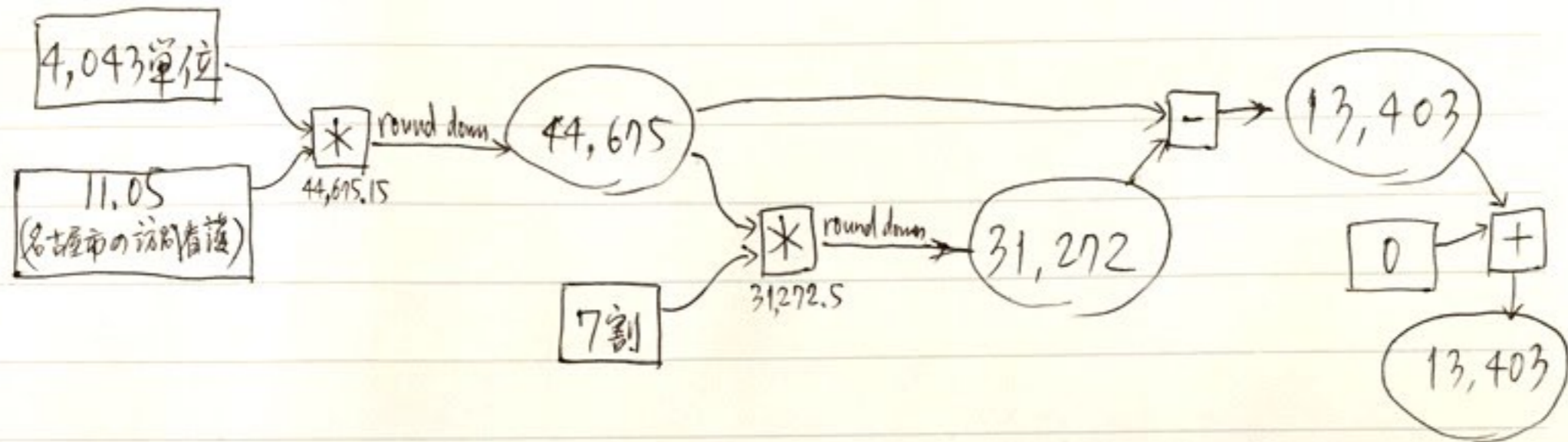
計算モデルとDSL

- C (=RASPでの構造化プログラミング)でプログラミングするのと、Haskell (=CBNの型付き λ 計算でのプログラミング)でプログラミングする時、見る世界が違う
- 問題ドメインが固定した場合、相性のいい計算モデルと相性の悪い計算モデルがある
- 例えば数式計算する時はスタックマシンか項書き換え系が相性がいいに対して、マシンラーニングの計算を表わすのに計算グラフが相性がいいのが感覚から分かる
- ということからDSL=ドメイン特化言語を作る時そのドメインにあう計算モデルを設計することがいいことだったりする

2flow

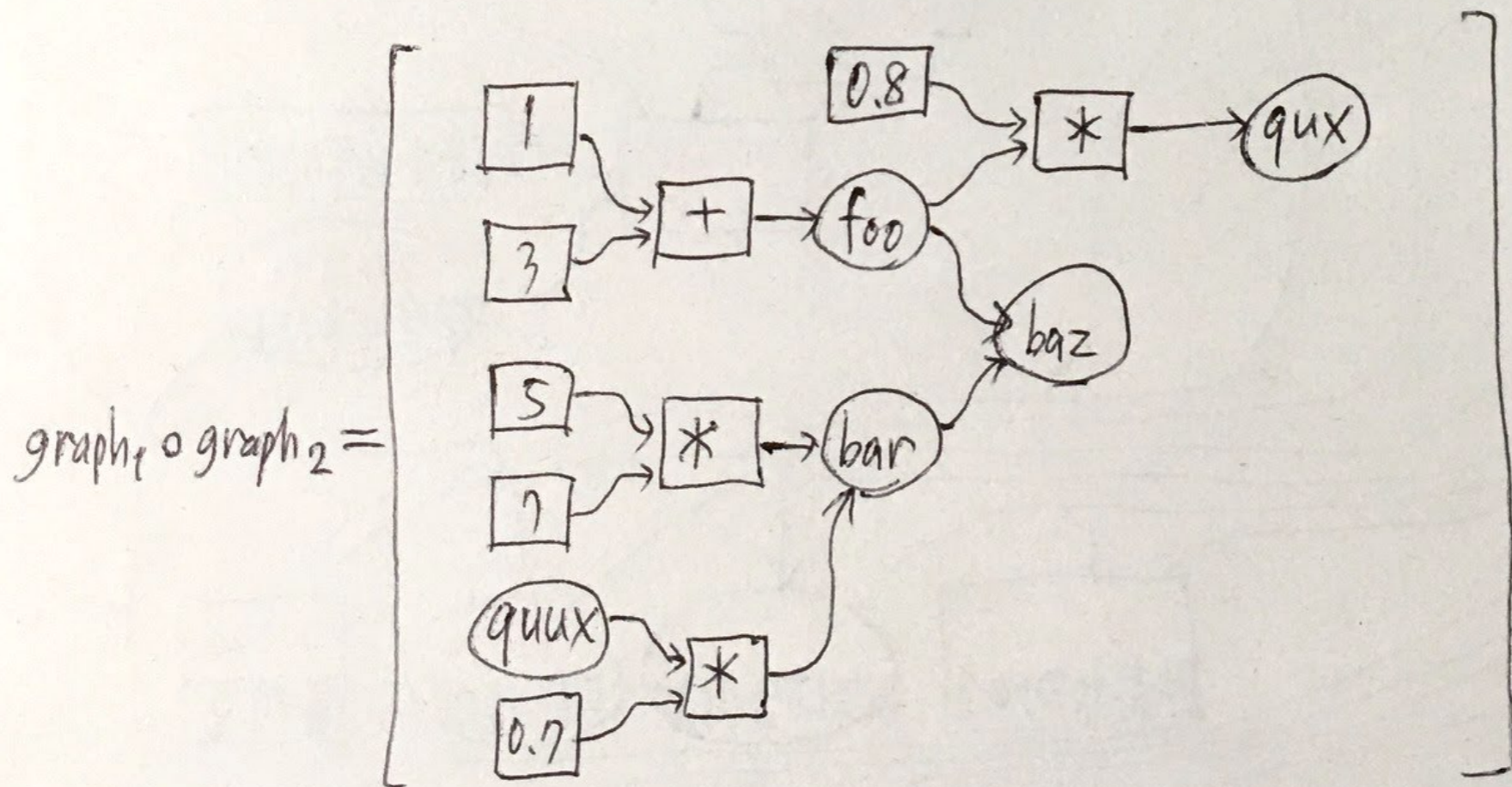
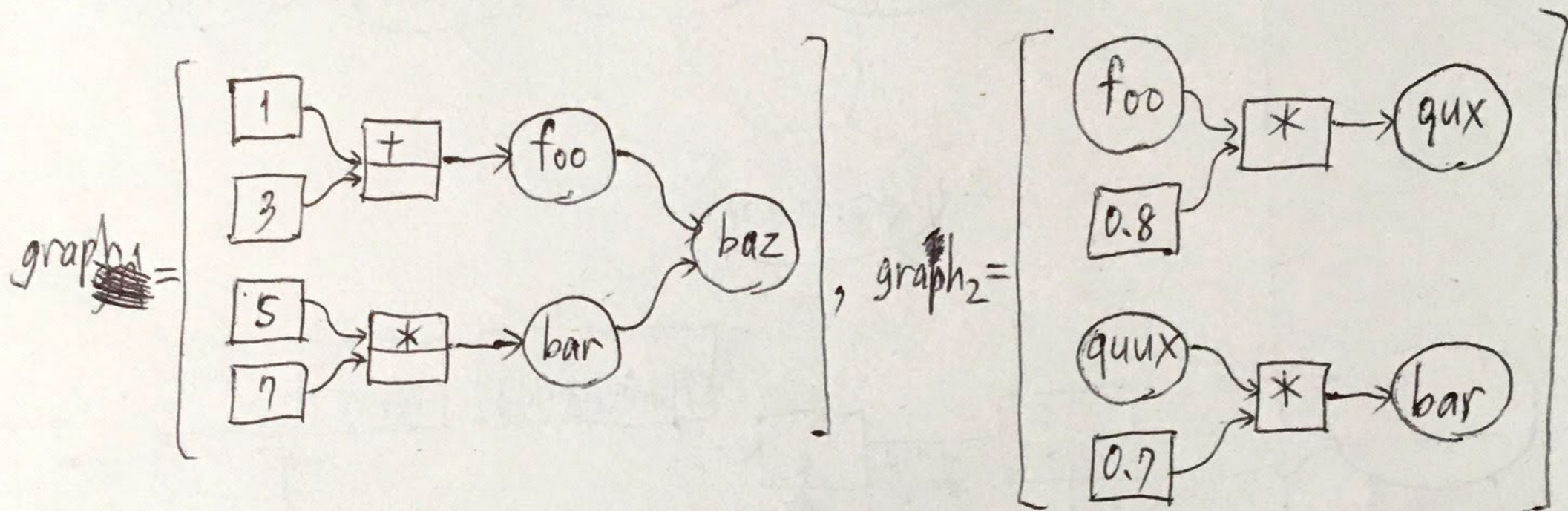
- 2flowはとあるDSLを作る時に設計された計算モデル
- 基本計算グラフである
- 但し複数の計算グラフを「いい感じに」くっつけてより多機能な計算グラフにできる → **Composability! Modularity!**

計算グラフ

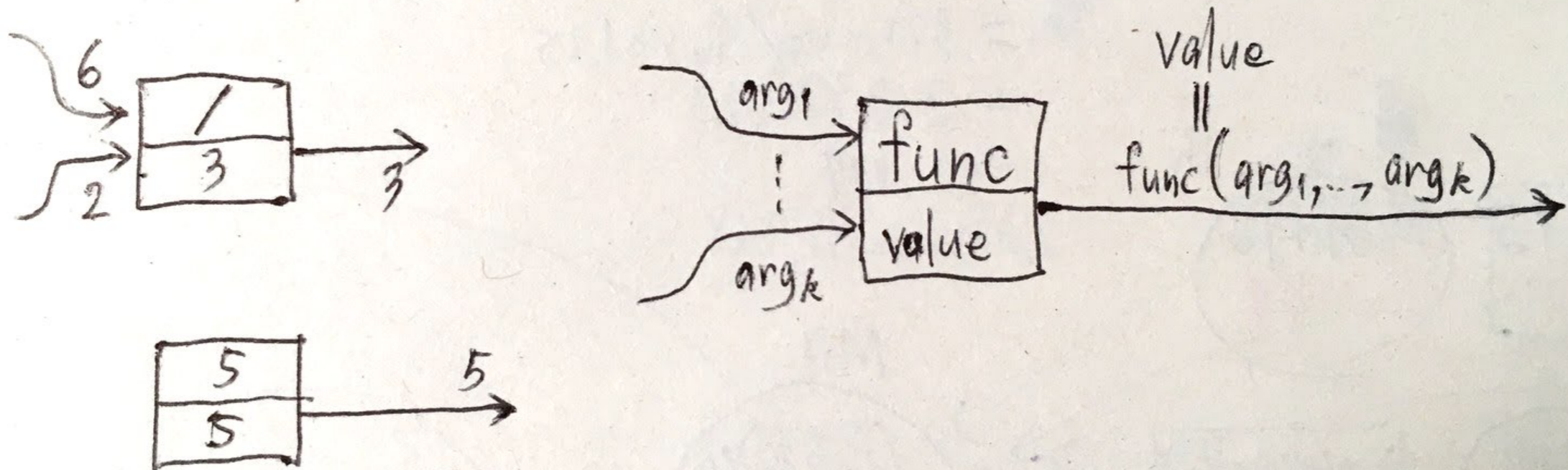


2flow

- 2flowはとあるDSLを作る時に設計された計算モデル
- 基本計算グラフである
- 但し複数の計算グラフを「いい感じに」くっつけてより多機能な計算グラフにできる
- くっつける = anode/fnodeとノードを二種類に分類し、グラフを合成するとき、同じ名前のanodeを同一視する

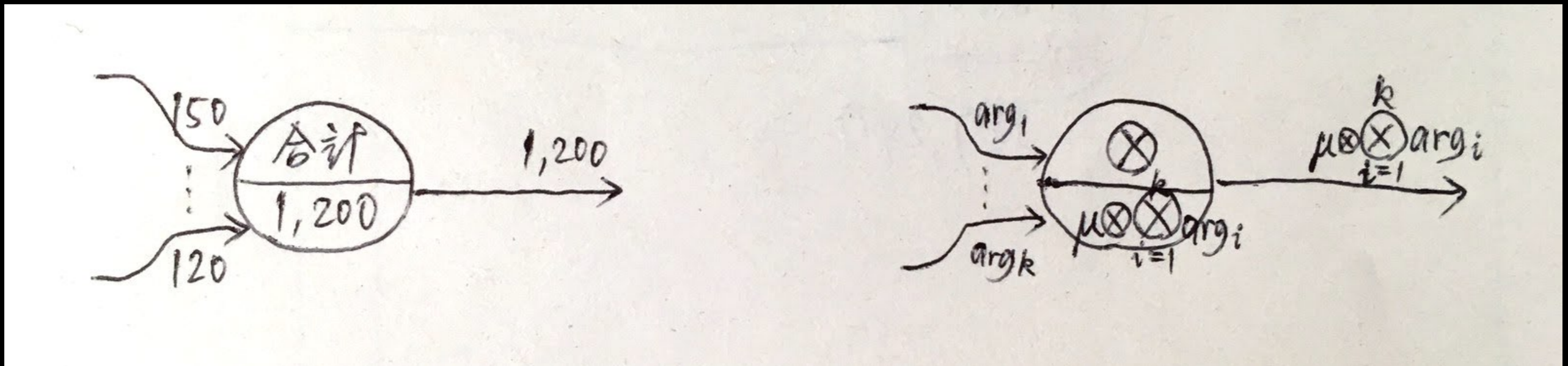


Composition (fnode)

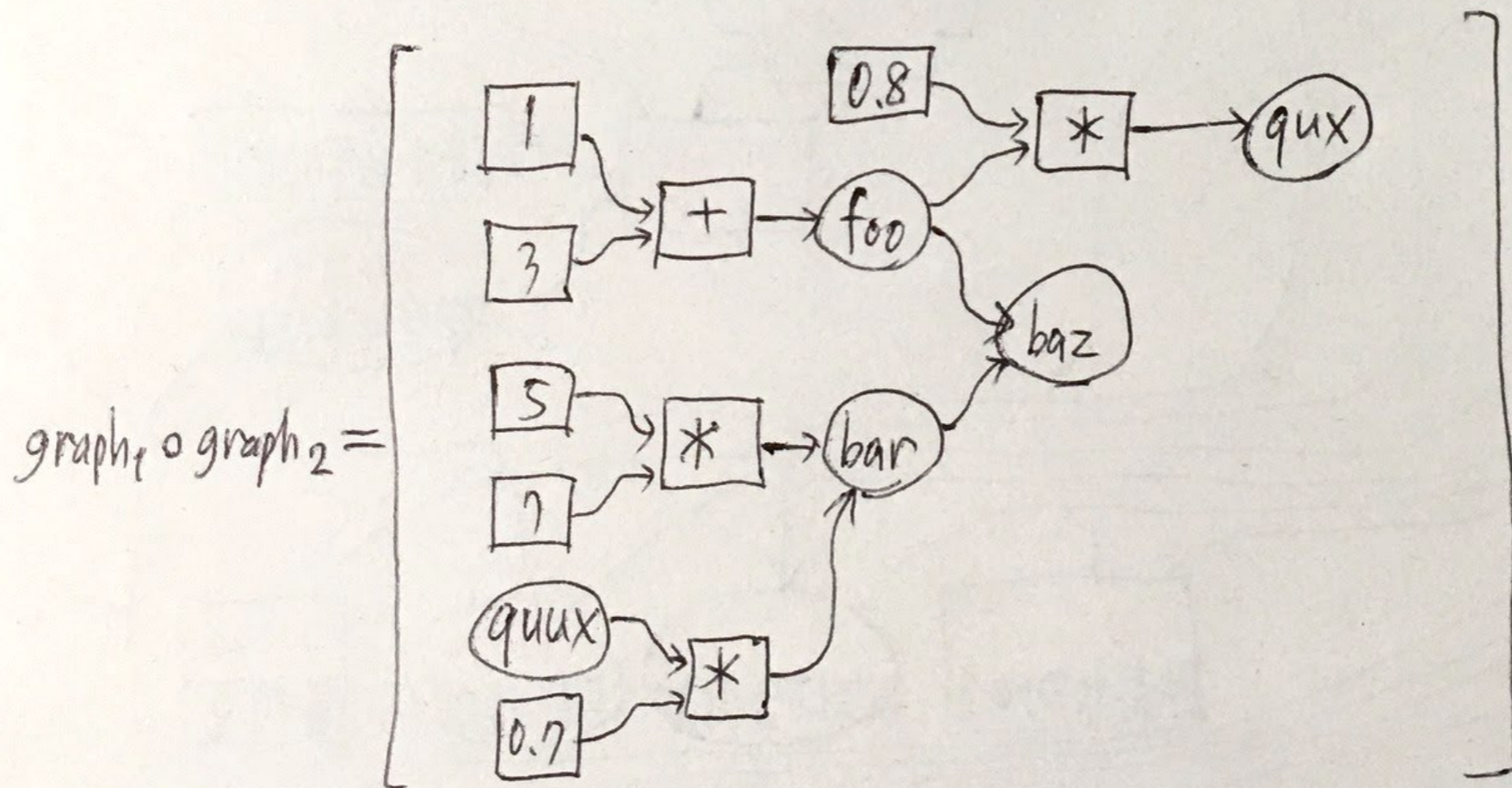
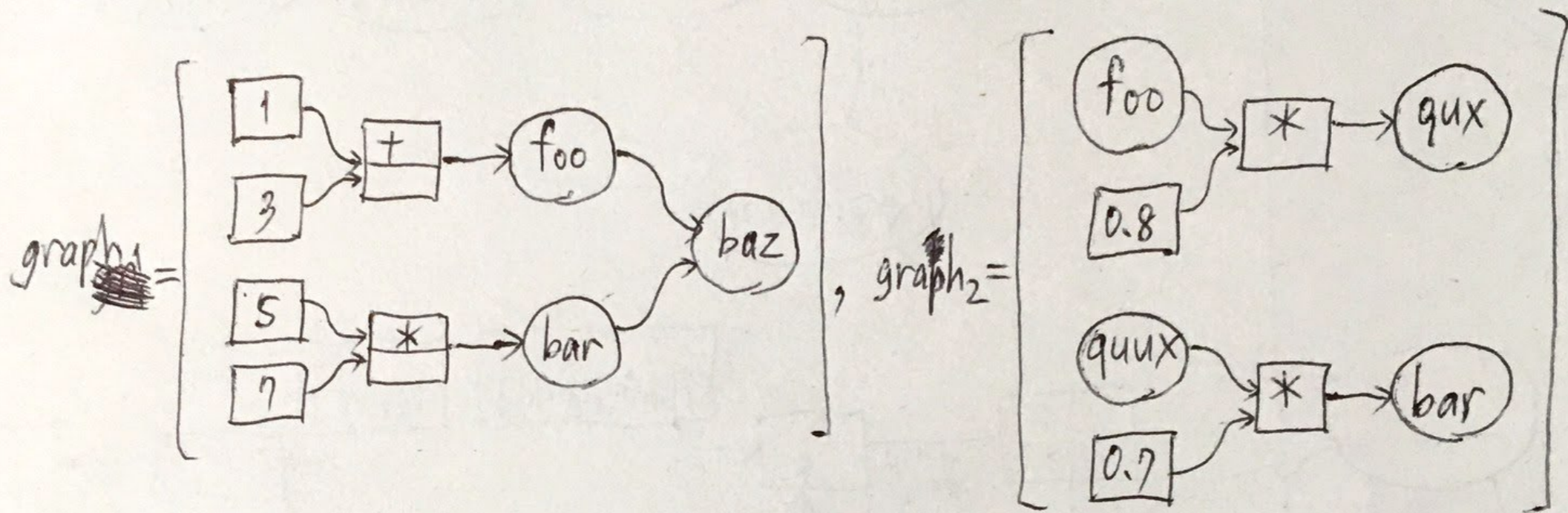


fnode: functional node

Composition (anode)



anode: accumulation node



何が嬉しい?

- 計算が非プログラマーにも簡単明快 (Scratchよりも遥かに!)
- 個々の機能を果すシンプルなグラフを有機かつ有意的に合成することでより多機能の計算グラフをシステムティックに作れる
- →つまり composability と local reasoning と modularity だ!
- 作って楽しい!

もっと知りたい

- Haochenに聞く
- PPLに来て!

ありがとうございました