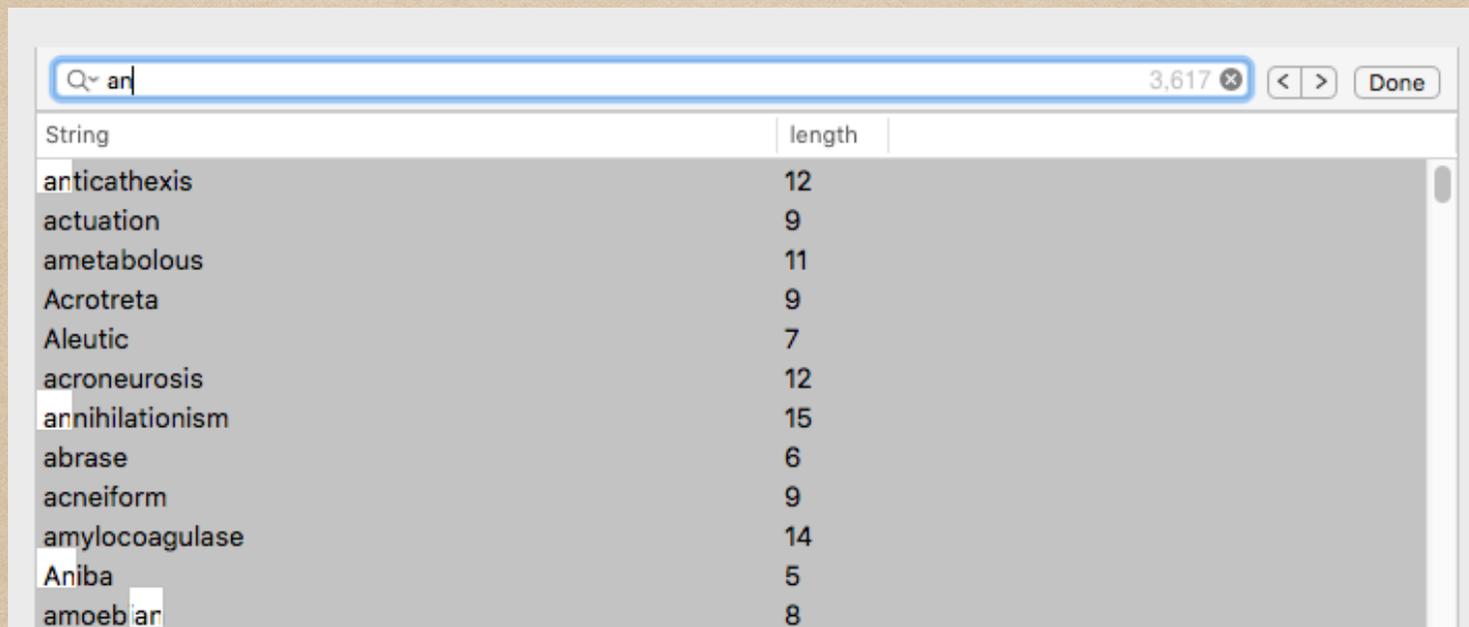


NSStringFinderの使い方

mindtools@mac.com

NSStringFinderとは何か

- ◆ UIで示すとこれ

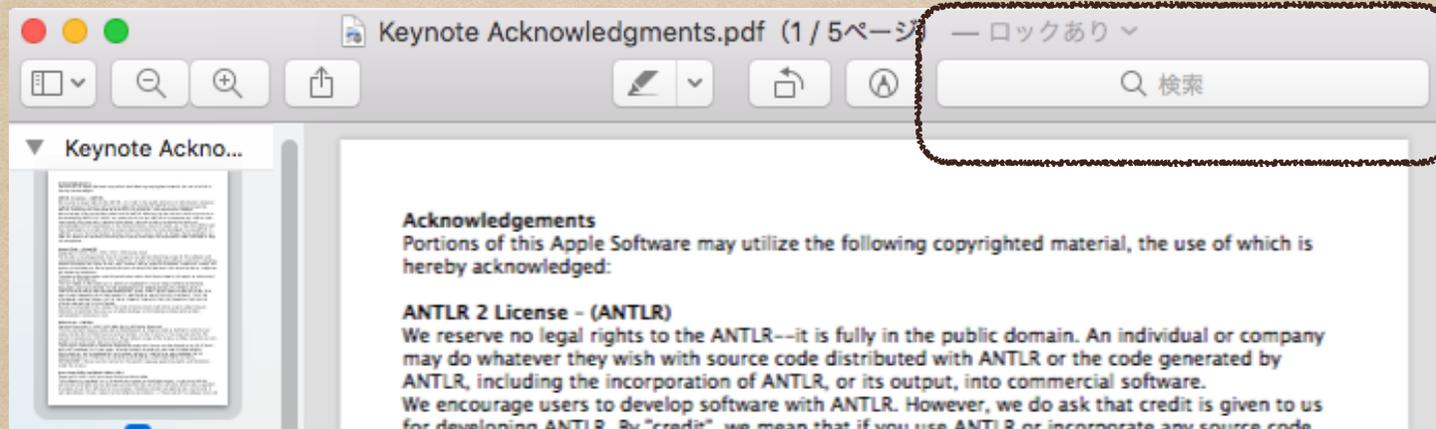


The screenshot shows the NSStringFinder application window. The search bar at the top contains the text 'an' and shows '3,617' results. Below the search bar is a table with two columns: 'String' and 'length'. The table lists various words starting with 'an' and their corresponding lengths.

String	length
anticathexis	12
actuation	9
ametabolous	11
Acrotreta	9
Aleutic	7
acroneurosis	12
annihilationism	15
abrase	6
acneiform	9
amylocoagulase	14
Aniba	5
amoebian	8

NSTextFinderとは何か

- ◆ ここじゃない



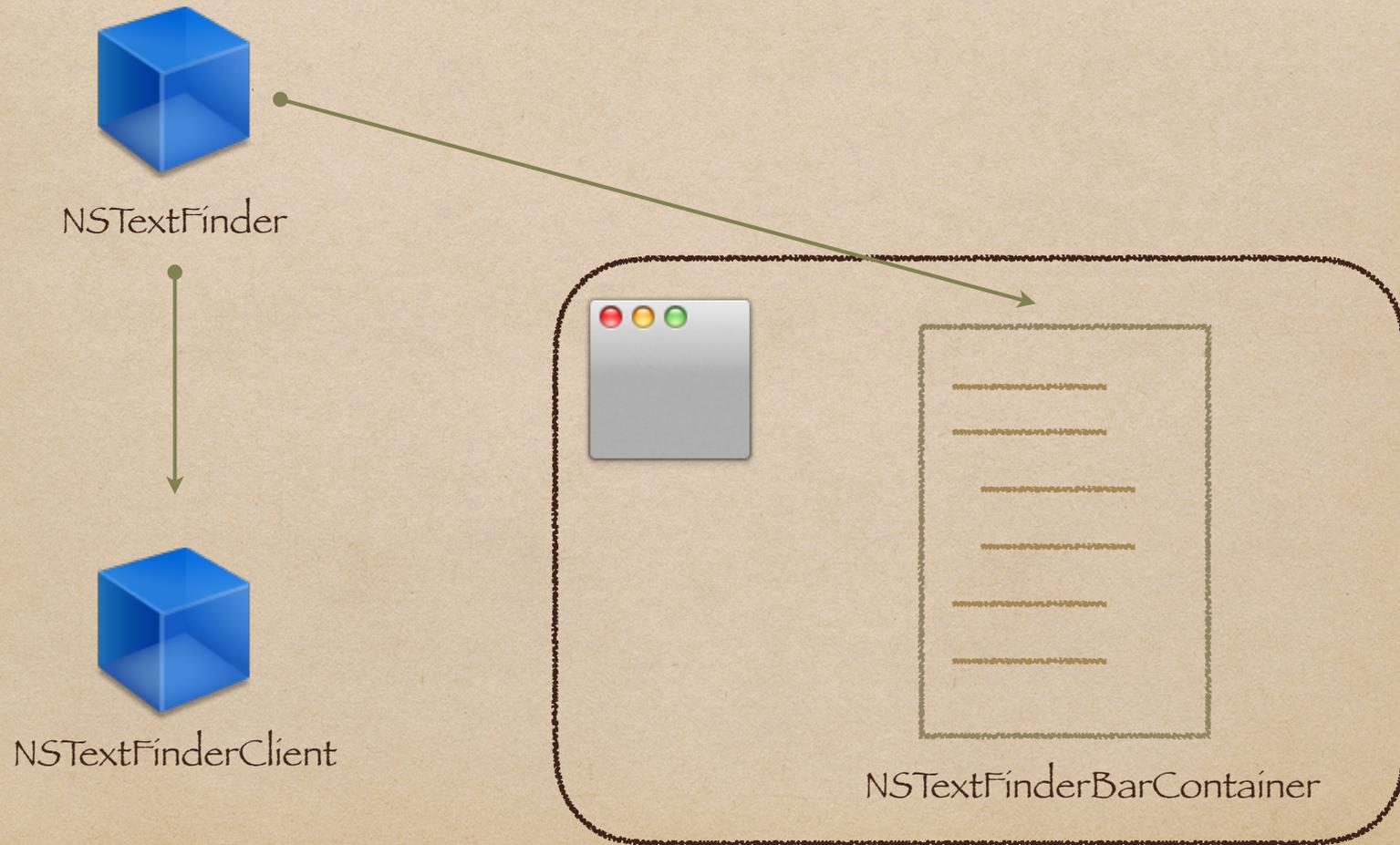
NSStringFinderとは何か

- ◆ フィルタ
 - ◆ 誰かが作った非常に多数のデータから目的の物を絞り込む
- ◆ ファインダ
 - ◆ 自分が作ったデータから目的の物を順次表示させる

インクリメンタルサーチ

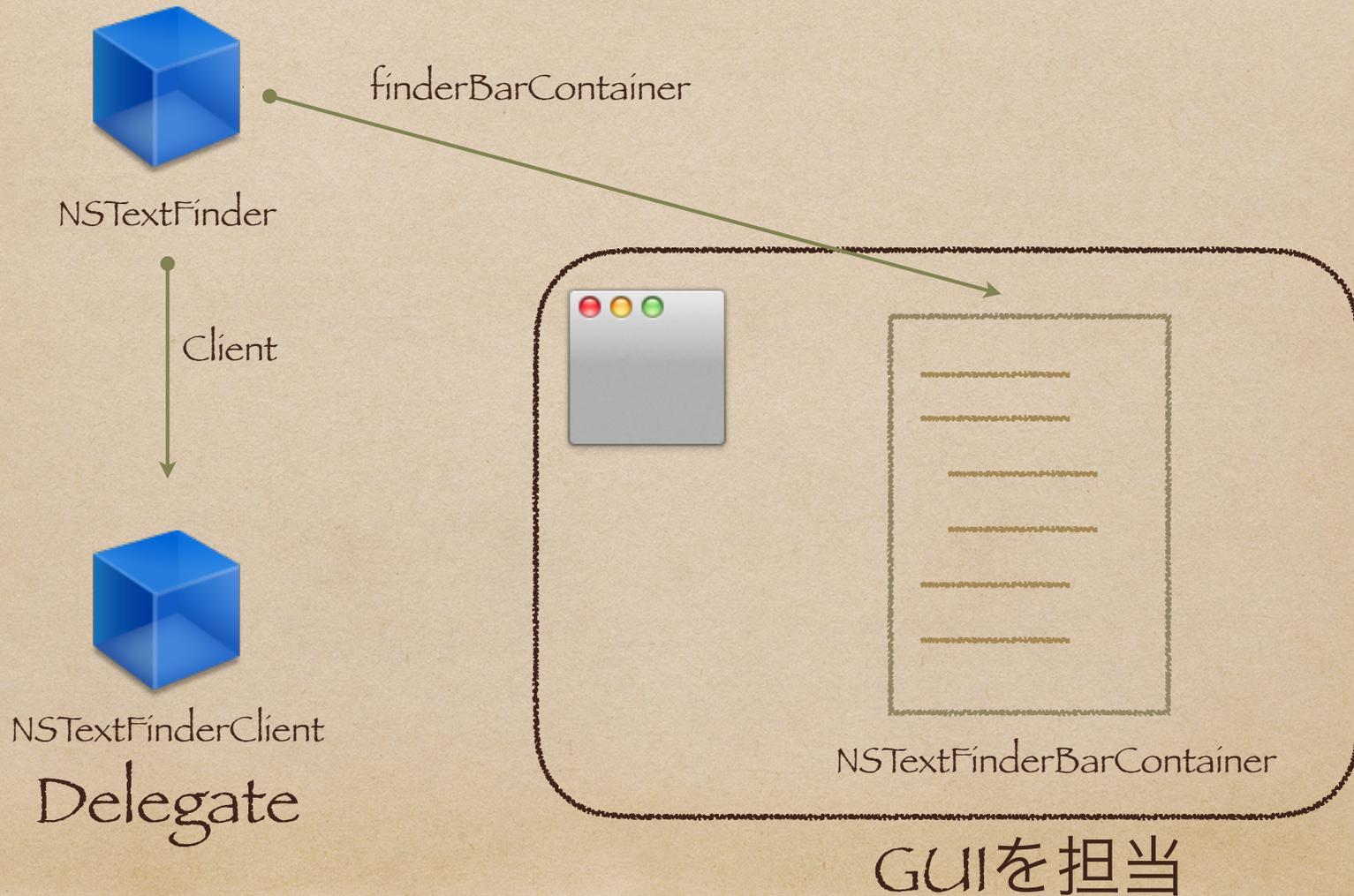
- ◆ 実装が面倒臭い、インクリメンタルサーチのロジックとUIを肩代わりしてくれる

NSTextFinderと協調する



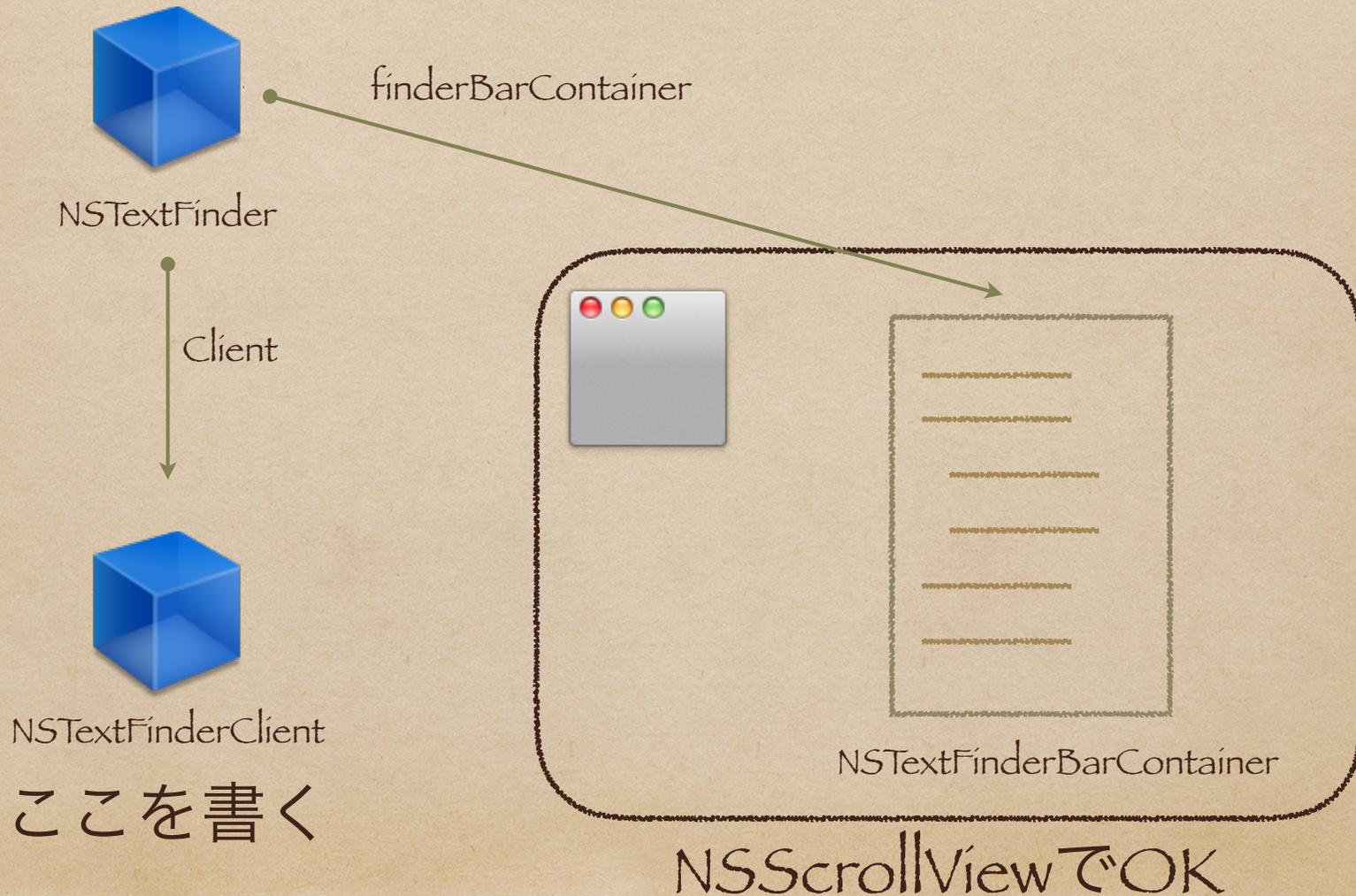
NSTextFinderと協調する

ロジック



NSTextFinderと協調する

Subclass化しない



NSTextFinderと協調する

NSTextFinderClientが提供するデータ構造

順
番
が
あ
る



テキストデータの塊



テキストデータの塊



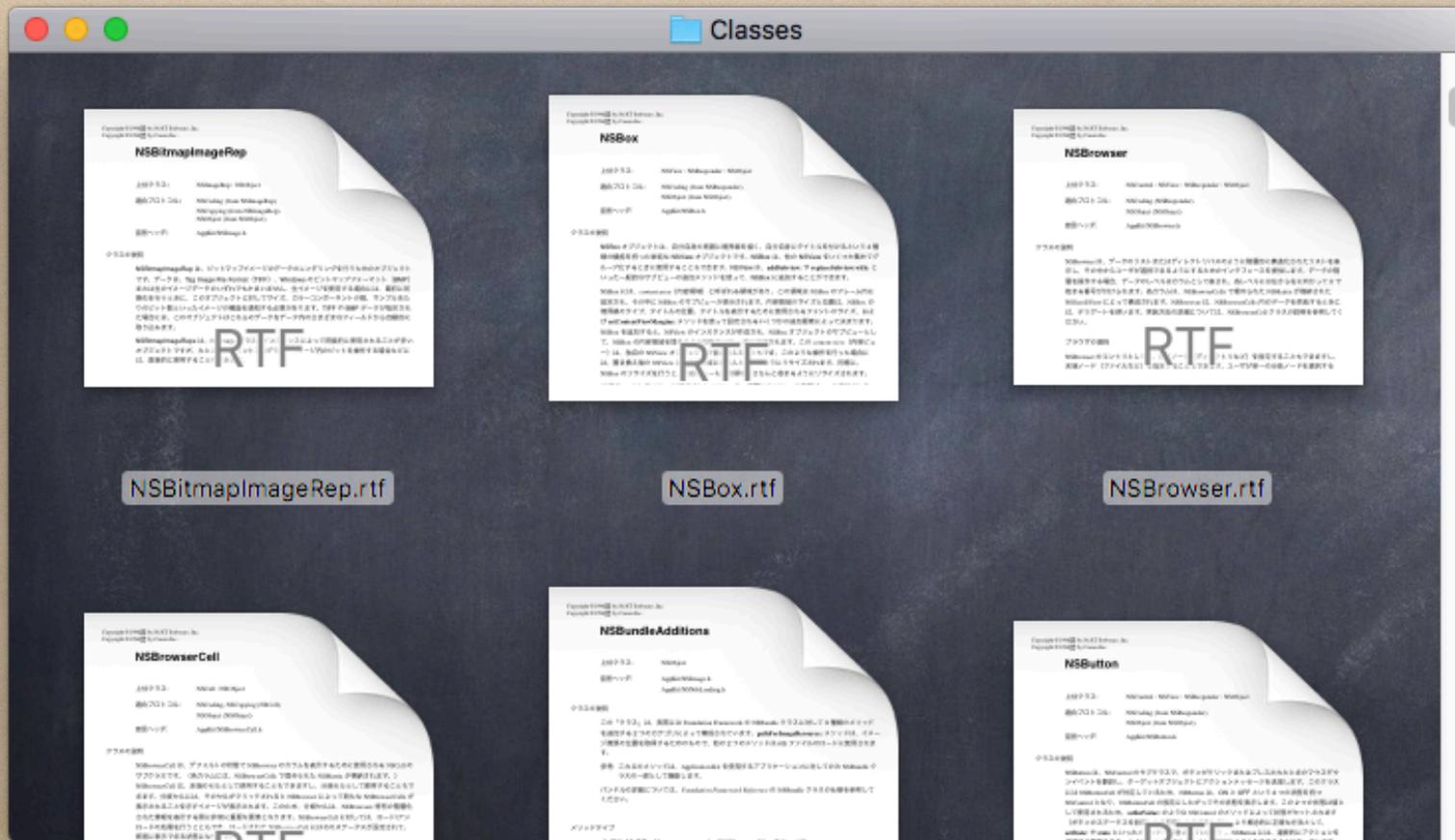
テキストデータの塊



テキストデータの塊

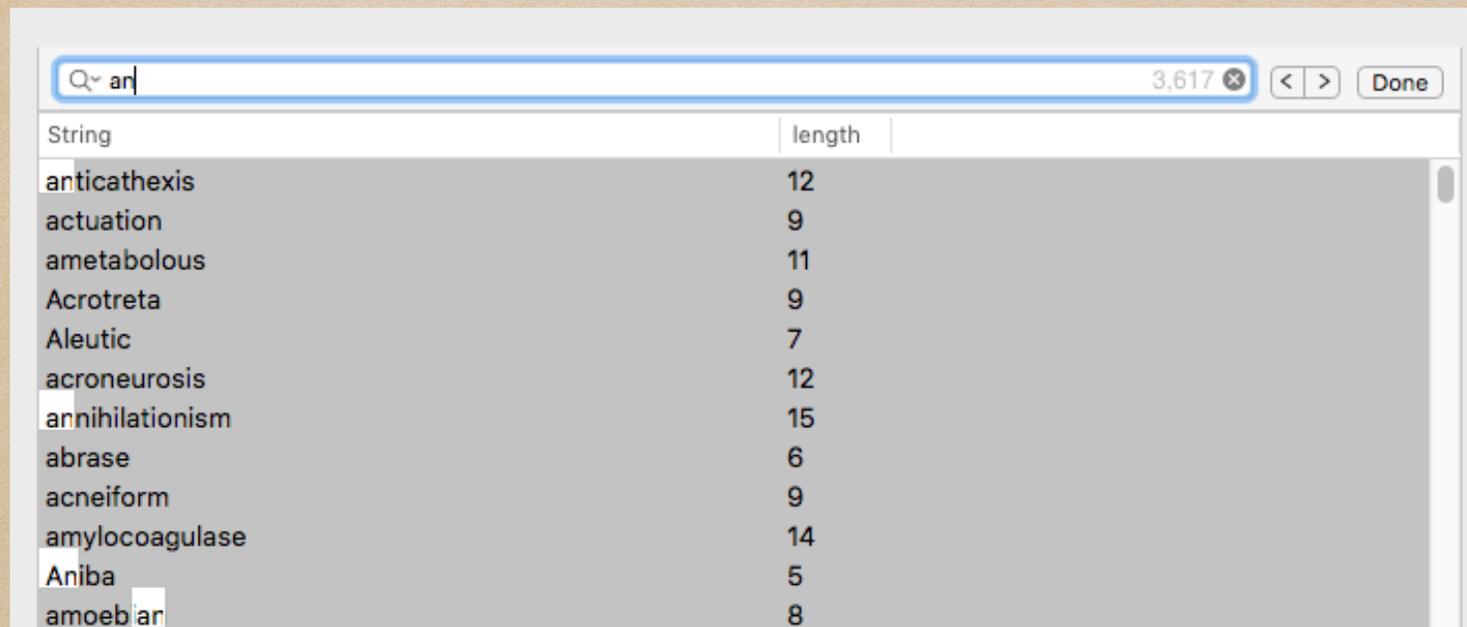
NSTextFinderと協調する

NSTextFinderClientが提供するデータ構造



NSStringFinderと協調する

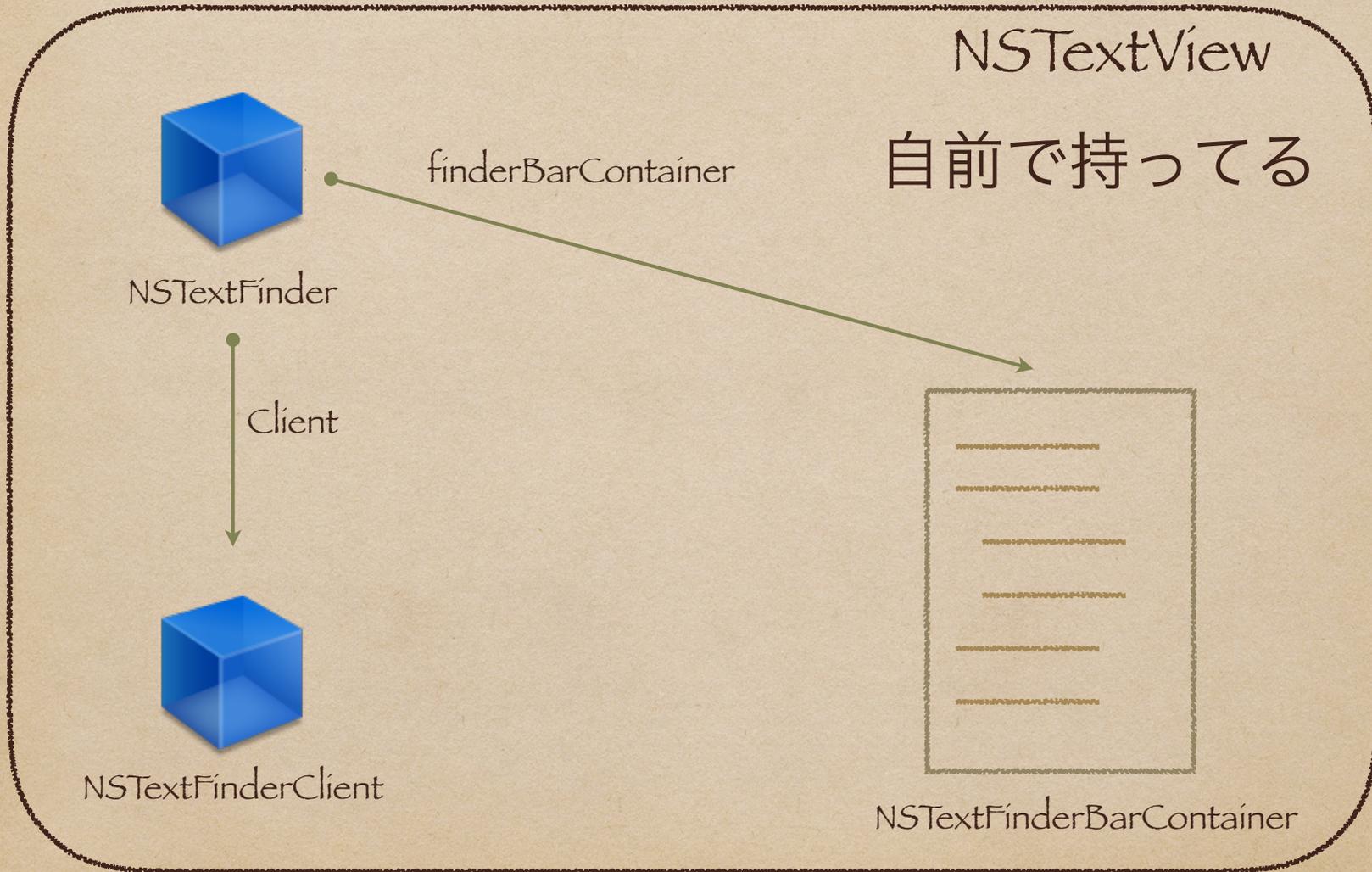
NSStringFinderClientが提供するデータ構造



The screenshot shows a search interface with a search bar containing the text "an". To the right of the search bar, it displays "3,617" results and navigation buttons for back, forward, and "Done". Below the search bar is a table with two columns: "String" and "length". The table lists several words and their corresponding lengths.

String	length
articathexis	12
actuation	9
ametabolous	11
Acrotreta	9
Aleutic	7
acroneurosis	12
annihilationism	15
abrase	6
acneiform	9
amylocoagulase	14
Aniba	5
amoebiar	8

NSTextViewでの使用例



NSScrollView

NSTableViewに適用する

Subclass化しない



NSTextFinder



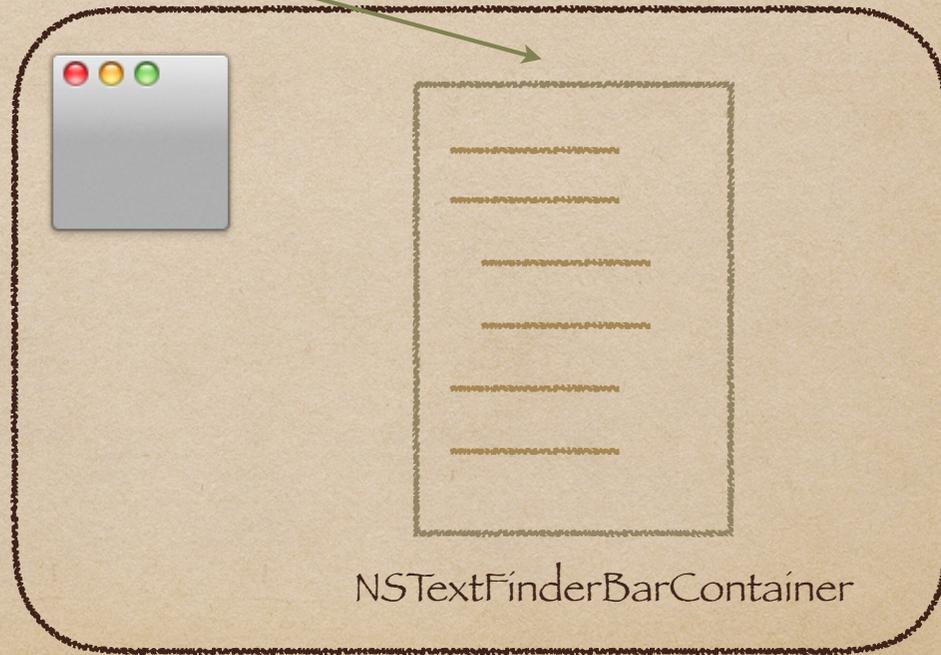
Client



NSTextFinderClient

ViewControllerに書く

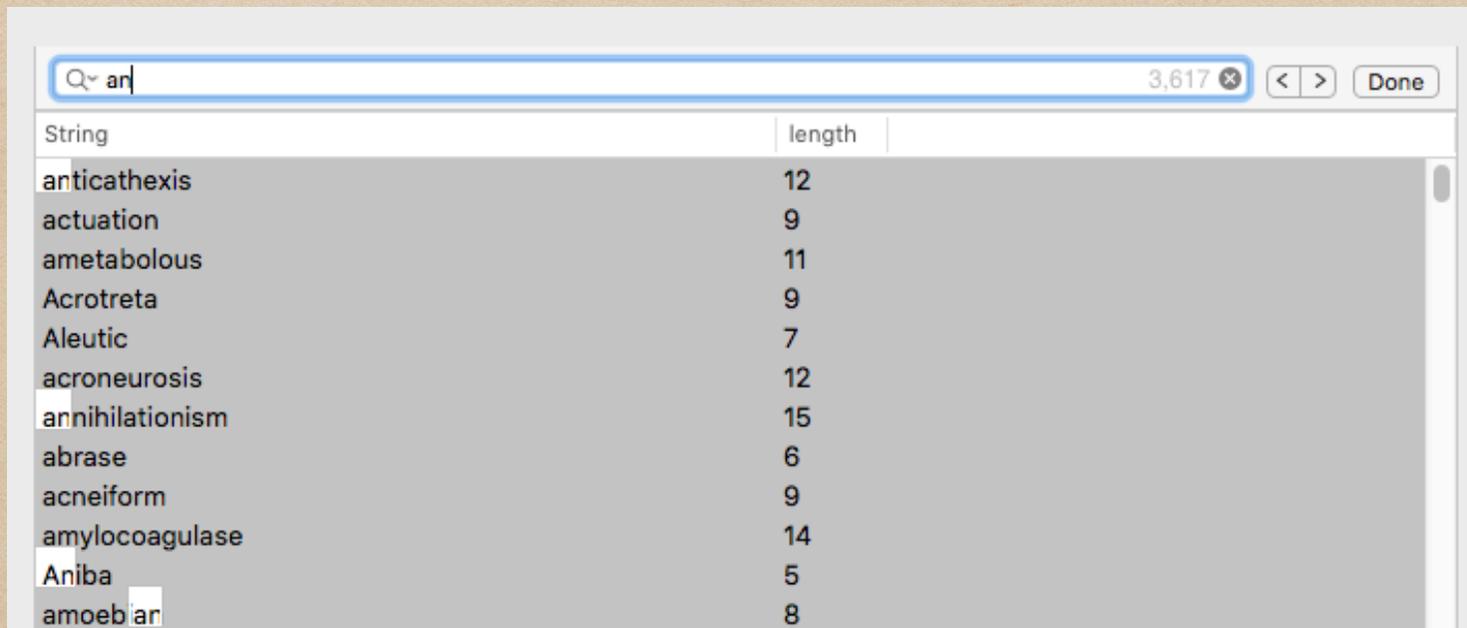
finderBarController



NSTextFinderBarController

NSScrollViewでOK

NSTableViewに適用する



Search results for 'an' (3,617 results):

String	length
anticathexis	12
actuation	9
ametabolous	11
Acrotreta	9
Aleutic	7
acroneurosis	12
annihilationism	15
abrase	6
acneiform	9
amylocoagulase	14
Aniba	5
amoebiar	8

サンプルコードの見どころ

ファイル一覧

TVFAppDelegate.swift	AppDelegateテスト用データを読み込んでいるだけ
TVFViewController.swift	今回のメインクラス。NSTableViewのdelegateであり、TextFinderのClient
TVFViewController+NSTextFieldClient.swift	追加したメソッド
words.txt	テスト用データ。/usr/share/dict/wordsのデータを間引き
Main.storyboard	findメニューのセレクトタを変更してます。

ファイル一覧

TableViewFinder_01がTextFinder適応前

TableViewFinder_02がTextFinder適応後



FileMergeで差分を見ながらコードを
読むとわかりやすいです

クラス一覧

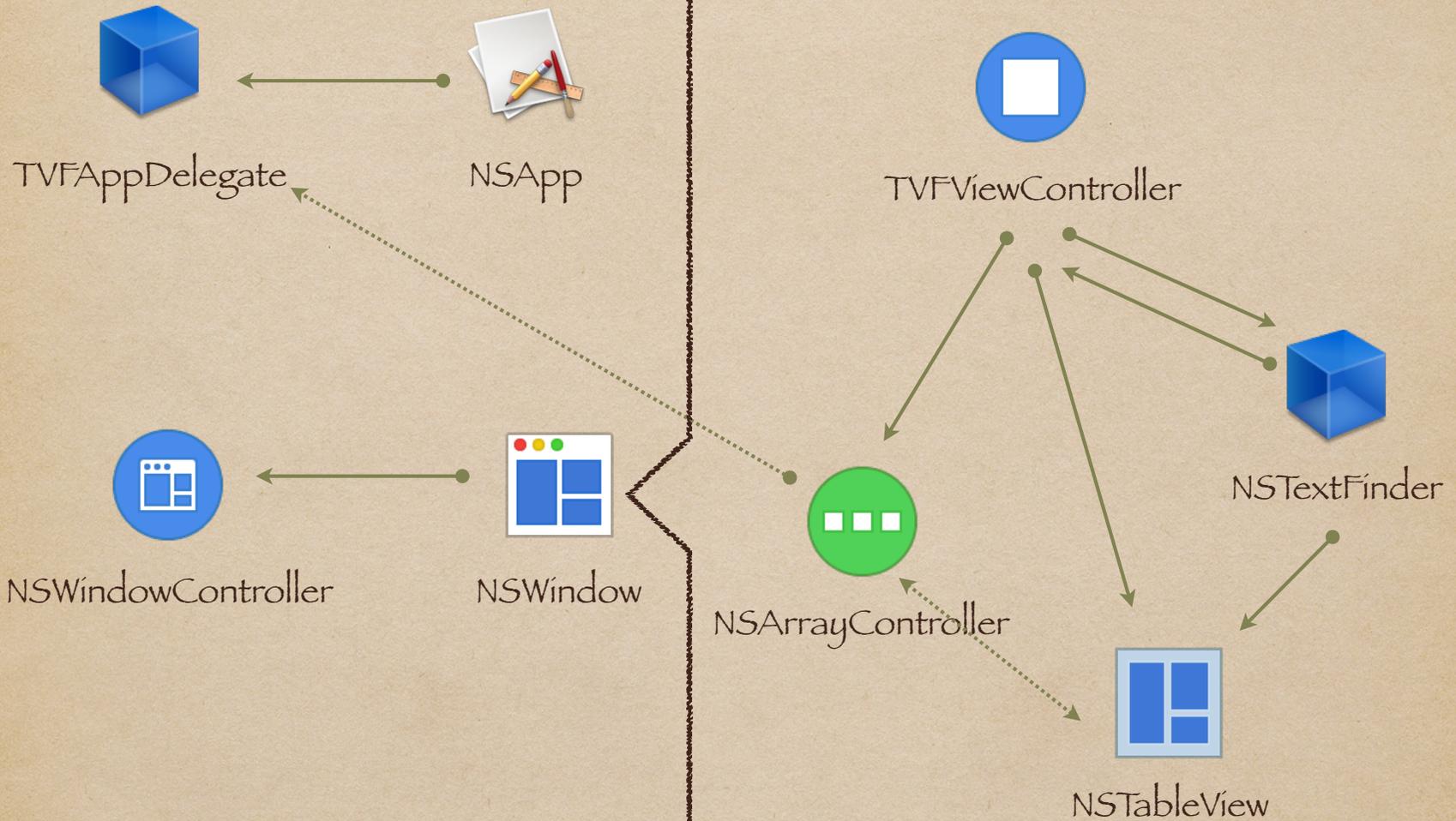
TVFAppDelegate.swift

テストデータを読み込んで
いるだけ。

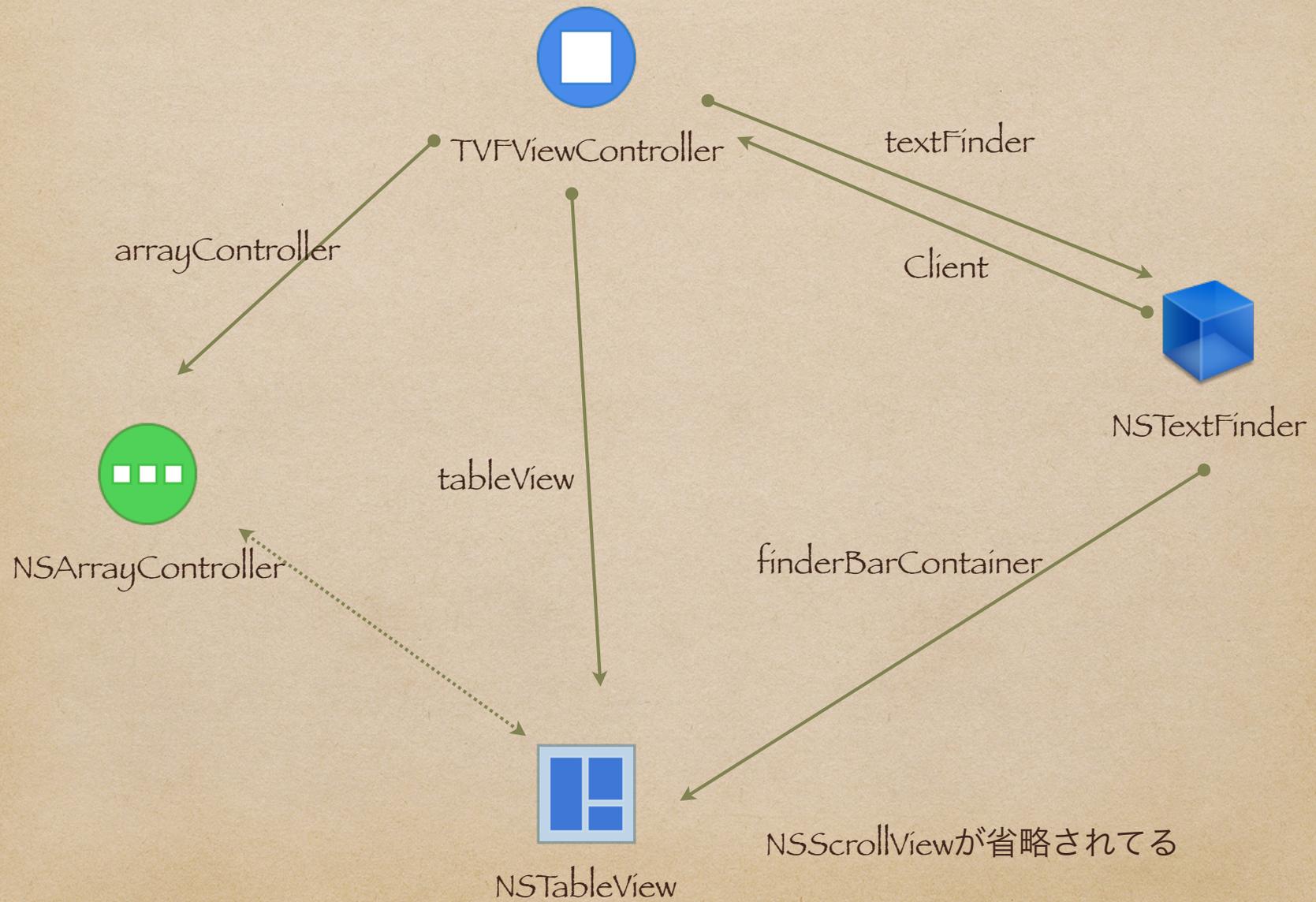
TVFViewController.swift

NSViewControllerのサブクラ
ス。NSTextFieldClientプロト
コルに適応。

オブジェクト図



オブジェクト図



テキスト検索データモデルの 移譲

部分文字列をクレ

```
public func string(at characterIndex: Int,  
                  effectiveRange outRange: NSRangePointer,  
                  endsWithSearchBoundary outFlag: UnsafeMutablePointer<ObjCBool>) -> String
```

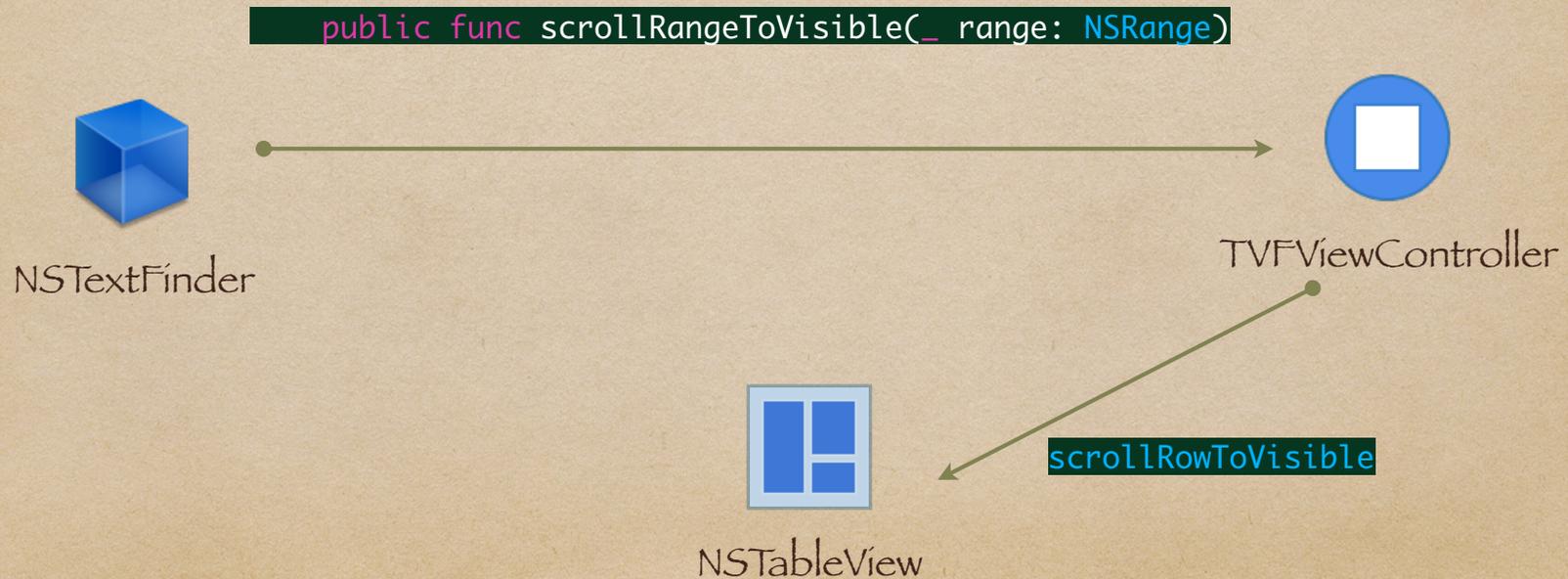


テキストデータの塊

subスレッドで動く

テキスト検索データモデルの 移譲

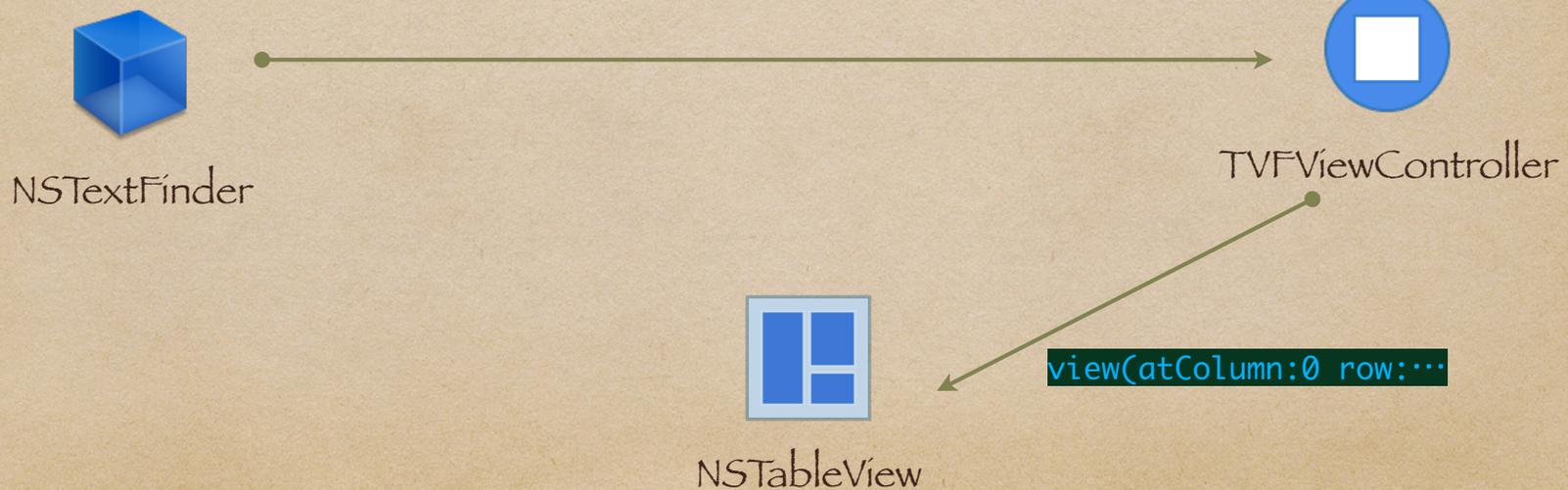
この部分までスクロールしてくれ



テキスト検索データモデルの 移譲

この範囲を含むViewを教えれ

```
public func contentView(at index: Int, effectiveCharacterRange outRange: NSRangePointer) -> NSView
```



テキスト検索データモデルの 移譲

この範囲を含む矩形を教えてください

```
public func rects(forCharacterRange range: NSRange) -> [NSValue]?
```



NSTextFinder



TVFViewController

テキスト検索データモデルの 移譲

描画される範囲だけ教えて。計算を楽しみたい

```
public var visibleCharacterRanges: [NSValue]
```



NSTextFinder



TVFViewController

実装の詳細へ

実装の見どころ (1)

```
override func viewDidLoad()
{
    super.viewDidLoad()

    // Do any additional setup after loading the view.

    self.textFinder = NSTextFinder()
    self.textFinder!.client = self
    self.textFinder!.findBarContainer = self.tableView!.enclosingScrollView

    self.textFinder!.isIncrementalSearchingEnabled = true
    self.textFinder!.incrementalSearchingShouldDimContentView = true

    // NSArrayControllerのarrangedArrayが変更された場合は、サーチ用の
    // データを再構築させるために、無効にする
    self.arrangedObjectsObservation = observe(\.arrayController?.arrangedObjects)
    {_,_ in
        self.setNeedUpdateFinderIndex()
    }
}
```

最初はインクリメンタルサーチをOFFにして実装する事

実装の見どころ (2)

```
override func viewDidLoad()
{
    super.viewDidLoad()

    // Do any additional setup after loading the view.

    self.textFinder = NSTextFinder()
    self.textFinder!.client = self
    self.textFinder!.findBarContainer = self.tableView!.enclosingScrollView

    self.textFinder!.isIncrementalSearchingEnabled = true
    self.textFinder!.incrementalSearchingShouldDimContentView = true

    // NSArrayControllerのarrangedArrayが変更された場合は、サーチ用の
    // データを再構築させるために、無効にする
    self.arrangedObjectsObservation = observe(\.arrayController?.arrangedObjects)
    {_,_ in
        self.setNeedUpdateFinderIndex()
    }
}
```

値が変更されると呼び出される。iOSでも使える。

書くのが簡単なKVO

実装の見どころ (3)

```
@IBAction override func performTextFinderAction(_ sender: Any?)
{
    if let theControl = sender as? NSValidatedUserInterfaceItem
    {
        let theAction: NSTextFinder.Action = NSTextFinder.Action(rawValue: theControl.tag)!
        self.textFinder!.performAction(theAction)
    }
}

func validateUserInterfaceItem(_ sender: Any?) -> Bool
{
    if let theControl = sender as? NSValidatedUserInterfaceItem
    {
        if theControl.action == #selector(NSResponder.performTextFinderAction(_:))
        {
            let theAction: NSTextFinder.Action = NSTextFinder.Action(rawValue: theControl.tag)!
            return self.textFinder!.validateAction(theAction)
        }
    }
    return false
}
```

MenuのActionの対応の仕方が標準的でない。

実装の見どころ (4)

```
private func calculateFinderIndex()
```

文字列	範囲	行
ABC	0~2	0
DE	3~4	1
FGHIJK	5~10	2
LMN	11~13	3

文字列の配列から

こっちの配列を作る

実装の見どころ (4)

```
func indexTupleFrom(location: Int) -> (range: Range<Int>, row: Int)?
```

文字の位置を与えると、

何行目にあるか、何文字目の範囲かを返す

このメソッドがあると、NSTextFinderの要求する

メソッドを書きやすくなる

参考文献

Appleのリファレンス

<https://developer.apple.com/documentation/appkit/nstextfinder>

NSTextFinder Magic

<https://blog.timschroeder.net/2012/01/12/nstextfinder-magic/>