

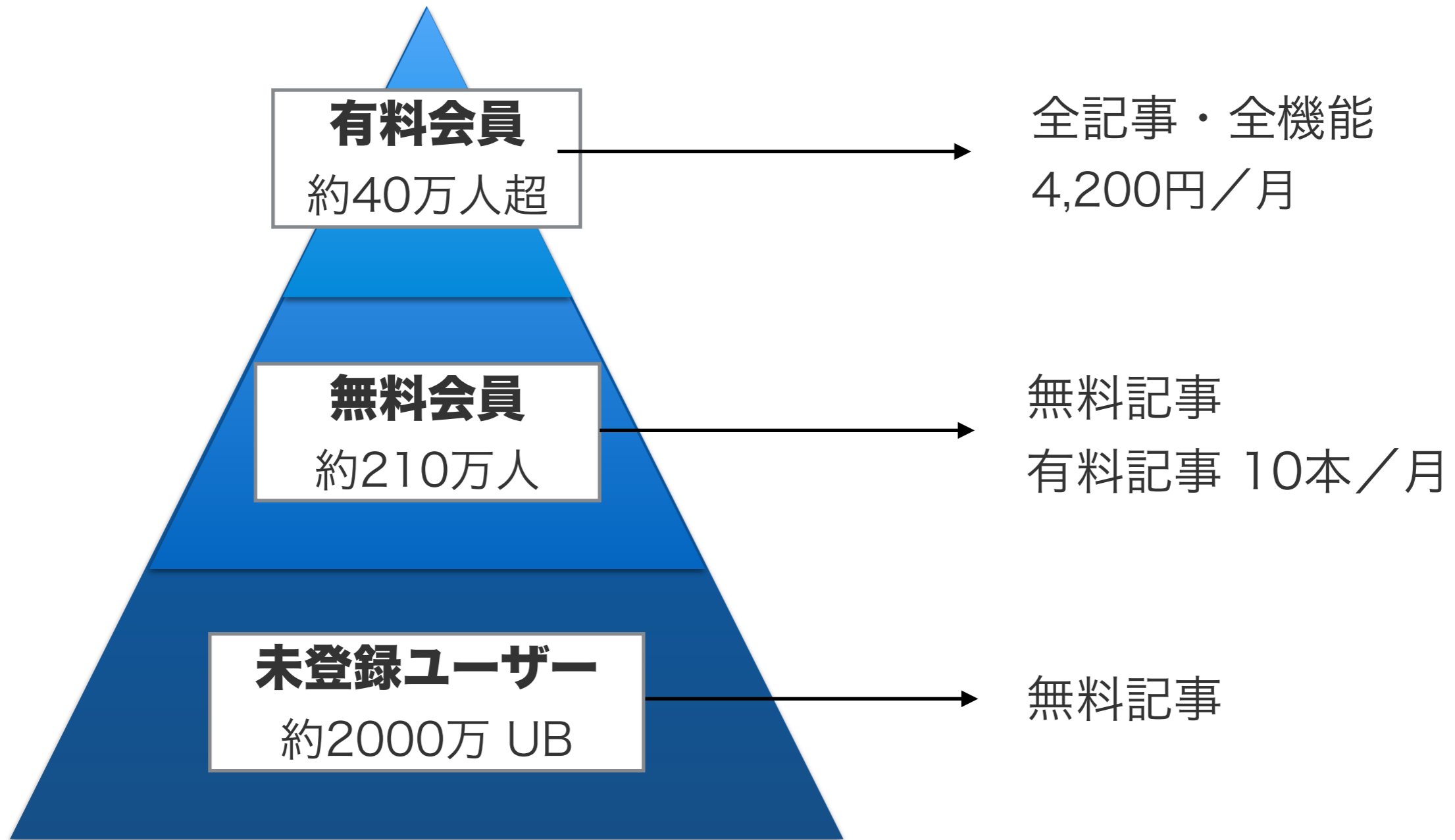
**日経電子版**  
**新聞アプリとしてのUI/UX**

**日本経済新聞社 武市 大志**

**2015/06/24**

- 大学卒業後、ガチでバンド活動。
- 2008年、Web開発会社に入社。Webエンジニアとして働く。FX取引のWebアプリ開発などに従事。
- 2014年、日本経済新聞社（本社）に入社。サービス開発のモバイルアプリチームに配属。





## 朝刊・夕刊

日本経済新聞のすべての記事

## Web刊 (=日経電子版)

Web限定の記事

## Myニュース

好きな連載や気になるキーワードをもとにパーソナライズ



# 日経電子版アプリをリニューアルしました

NIKKEI

- UI/UXの大幅な刷新
- サーバサイドをAWSへ
- iOSのフロントサイドを内製化  
(Androidも内製に移行中)



- **新聞アプリのUI/UX**
- **リニューアルって胃が痛い**
- **サクサク感を出すための実装**

# 新聞アプリのUI/UX

新聞アプリのUI/UX  
膨大な記事とカテゴリ

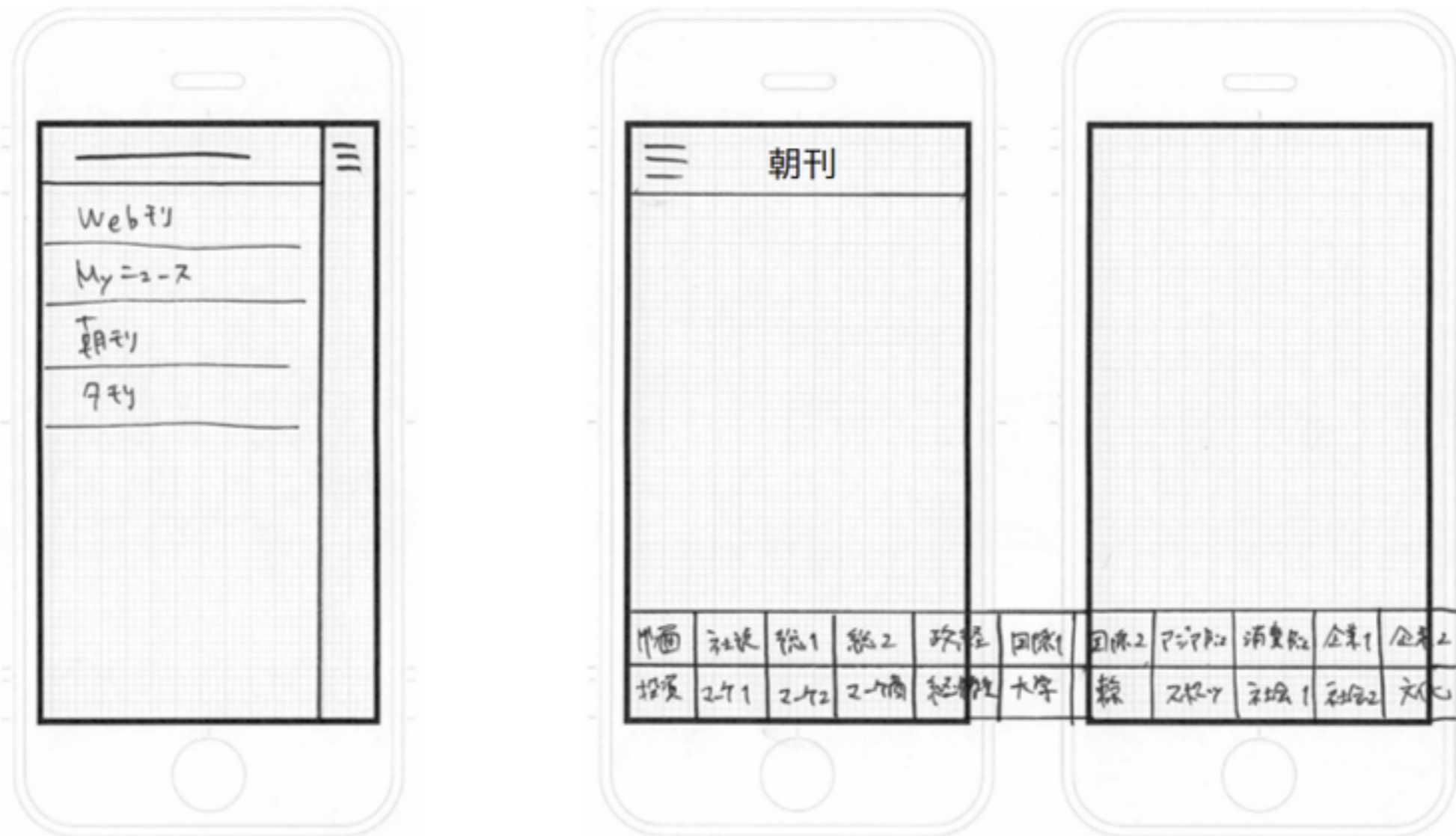


## 最大の人気コンテンツ

- ・ 約40ページ
- ・ 約300本の記事
- ・ 文字数は約15万文字  
= 新書2冊分！！



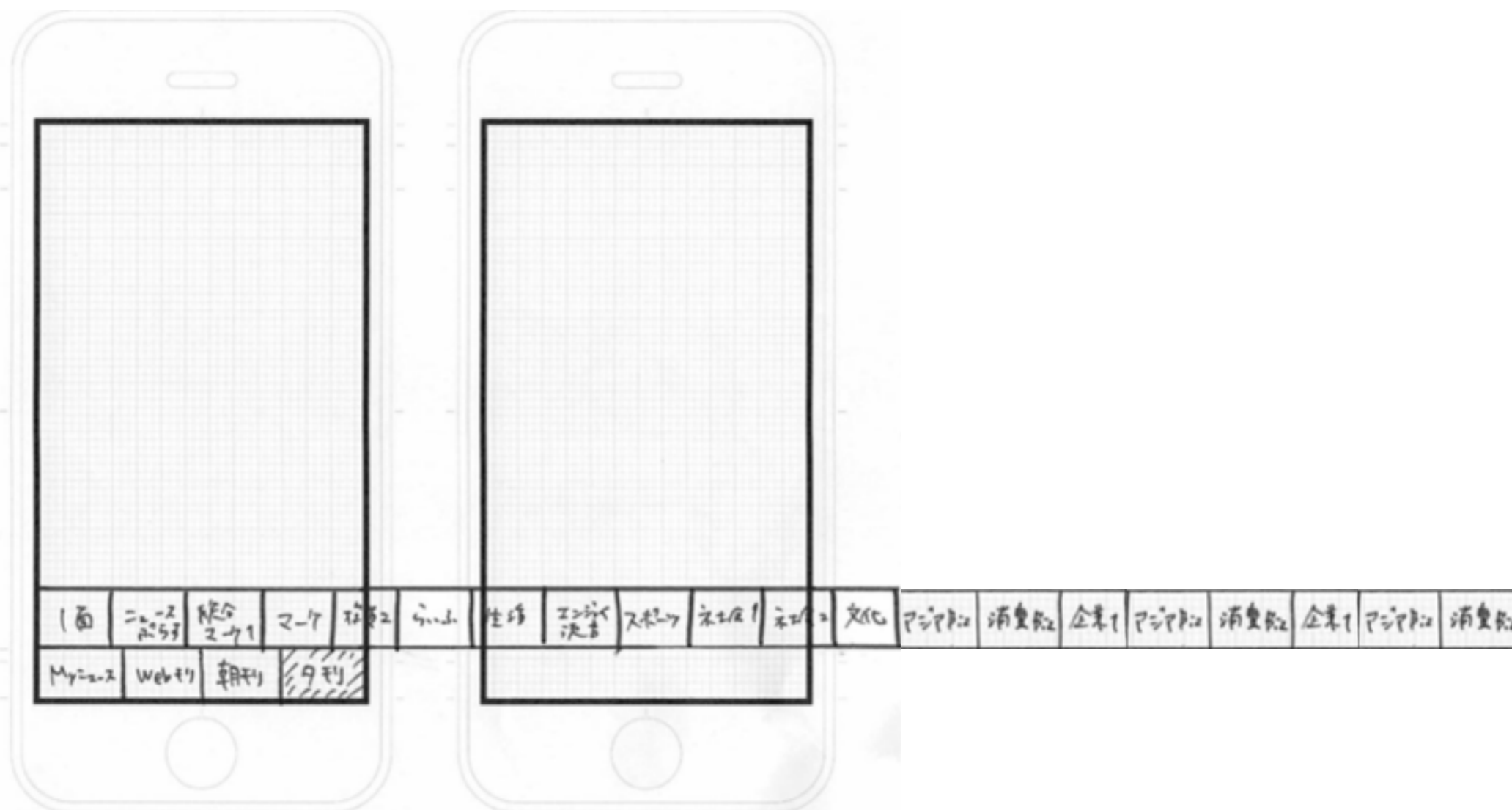
## スライドメニュー (2段)



メリット : すべての面が2.5画面分の横スクロールでおさまる

デメリット : Web刊 ⇔ 朝刊・夕刊 ⇔ Myニュースの回遊性が悪くなる

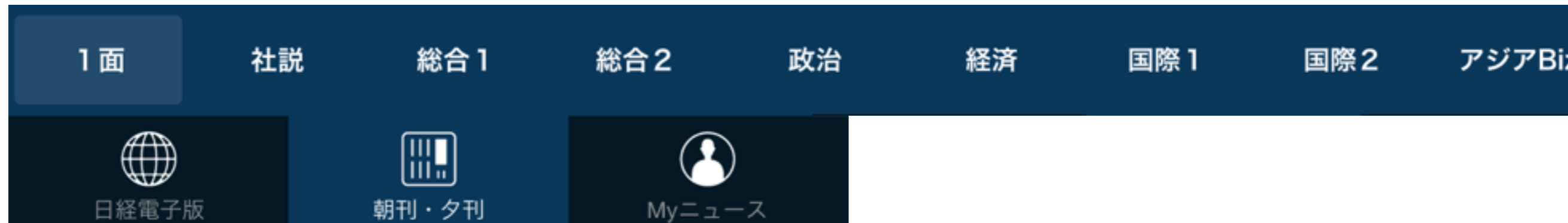
## タブメニュー + スライドメニュー (1段)



デメリット：終わりの面まで4.5画面分の横スクロールが必要になる

メリット：Web刊 ⇔ 朝刊・夕刊 ⇔ Myニュースの回遊性が良い

## タブメニュー + スライドメニュー（1段）を採用



- ・ 先頭に近い面がよく読まれていた
- ・ Web刊 ⇔ 朝刊・夕刊 ⇔ Myニュースの回遊性を重視
- ・ 本当はスライドメニューに表示する面を、  
ユーザーがカスタマイズできるようにしたかった・・・

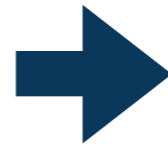
新聞アプリのUI/UX

PC最適であるがゆえの深い階層構造

## Before

朝刊・夕刊 2日分

Web刊 トップの10本  
速報



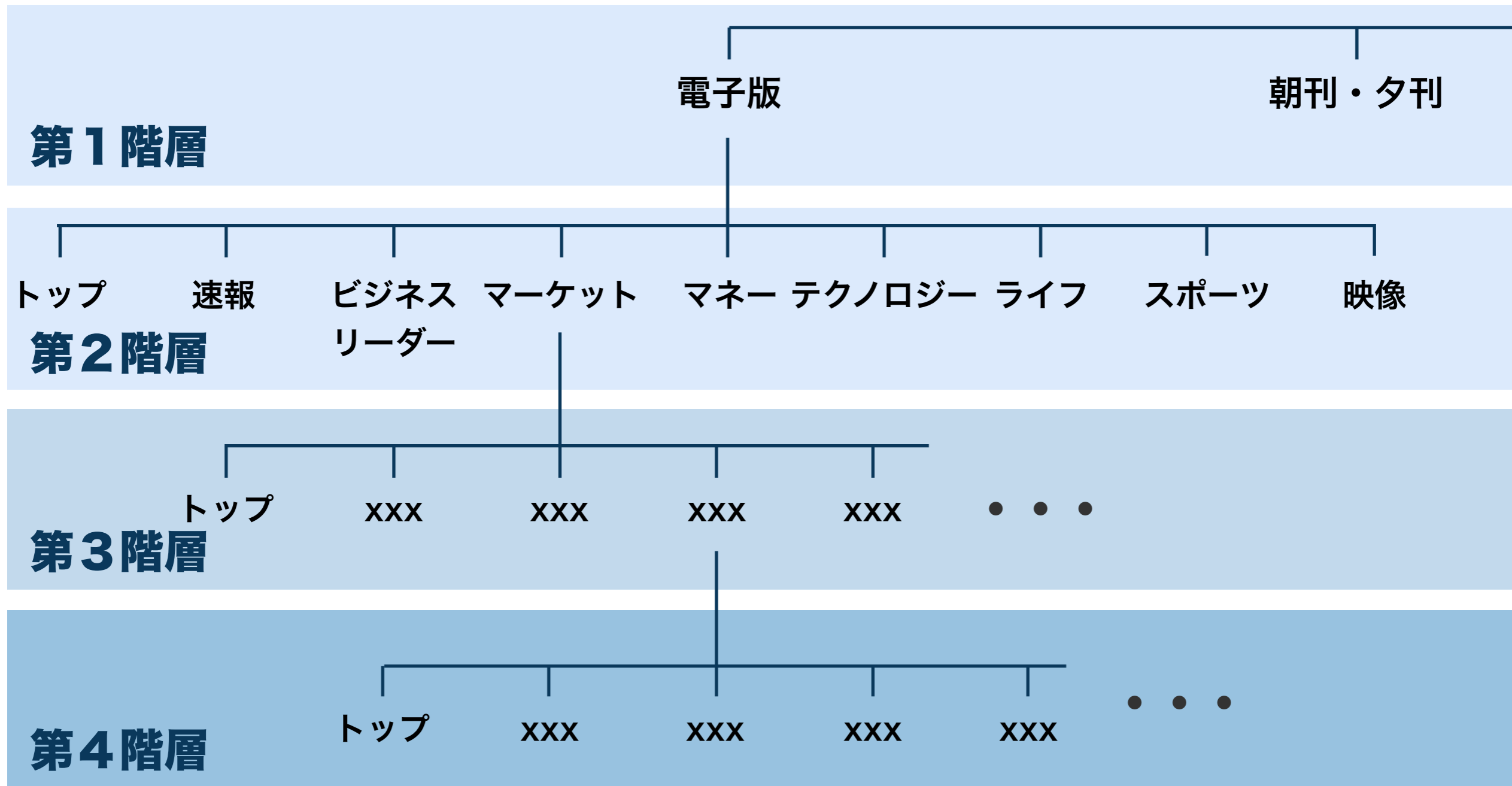
## After

朝刊・夕刊 1週間分

Web刊  
ビジネスリーダー／マーケット  
テクノロジー／マネー／ライフ  
スポーツ／映像  
速報

検索  
企業情報  
用語説明

PCサイトに由来した深い階層構造が、もれなくついてくる。





**無理やり表現しても、使いこなせない。  
むしろ混乱するだけ。  
供給者側の自己満足になる。**

- ・ 第1、第2階層までをしっかりと表現。
- ・ 第3階層は最悪使われなくてもいい。
- ・ 第4階層は表現しない。

本当はシンプルな構造に変えたかった・・・



# そのかわり第1・第2階層の回遊性をちゃんとあげる

NIKKEI



スクロール中は邪魔にならないよう  
メニューが消える。



下まで来たら、メニューが現れる。  
「次は何を読もう」

新聞アプリのUI/UX

ユーザー層によってまったく異なる使い方

## 記事一覧



## 個別記事

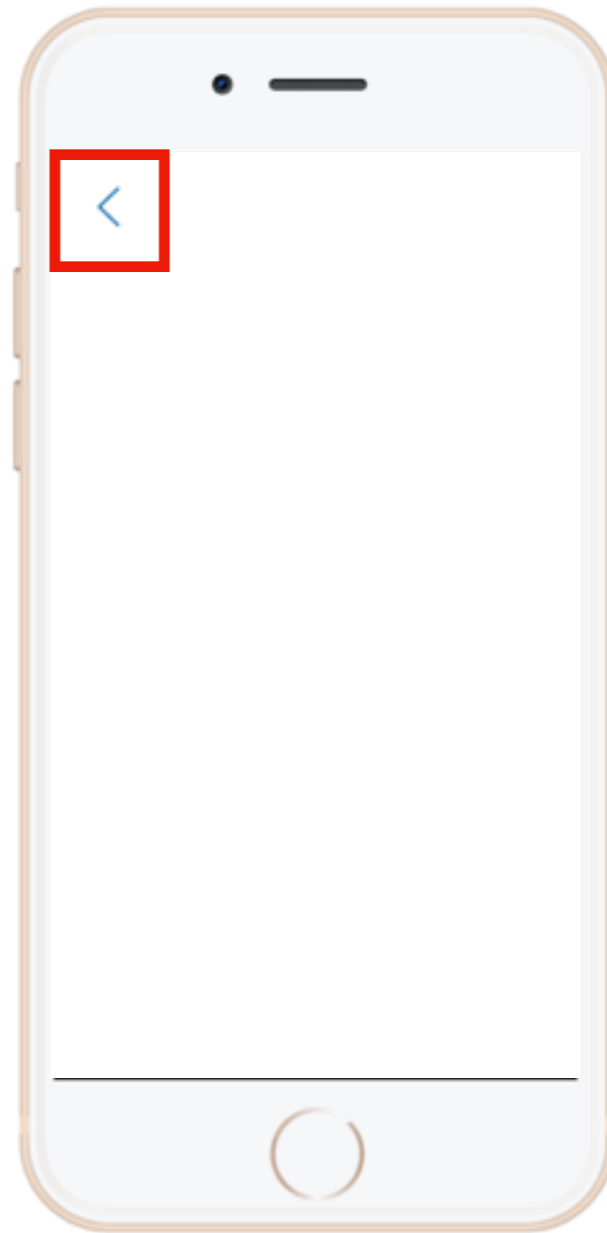
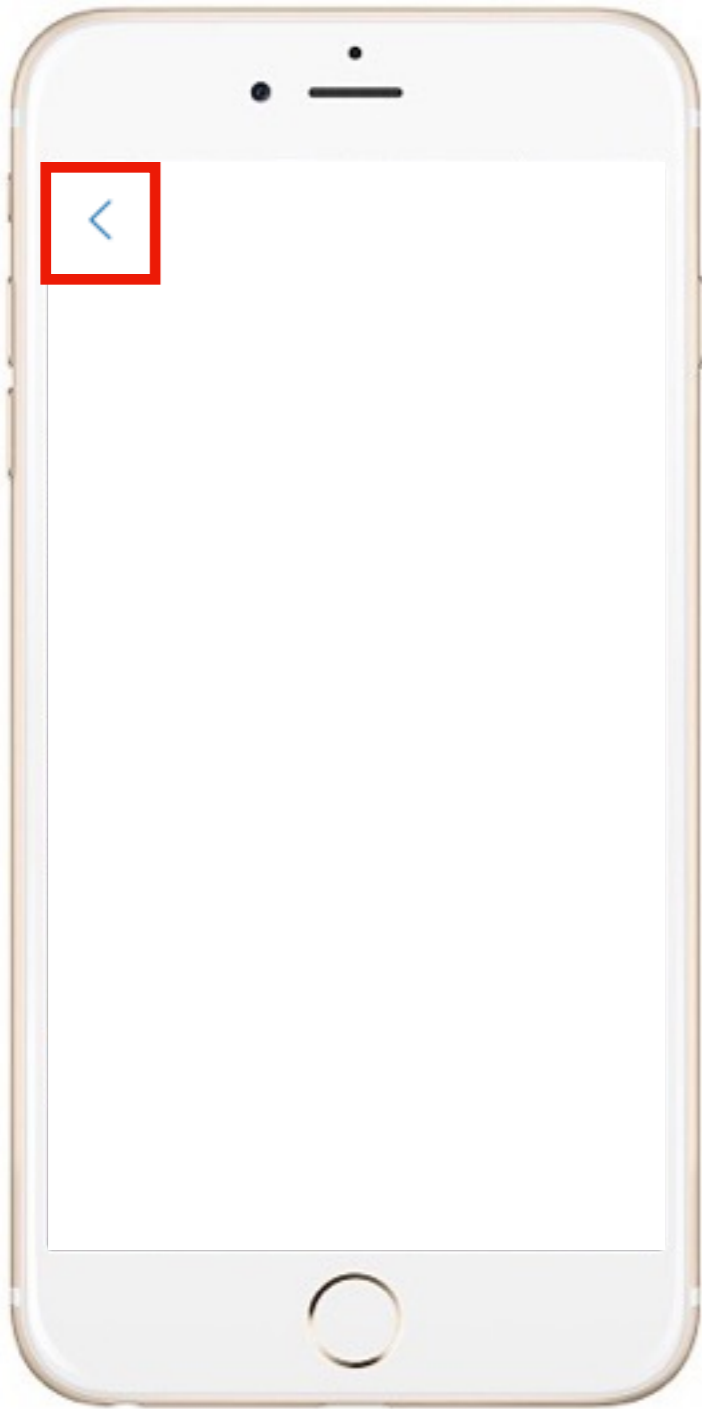


## 記事一覧

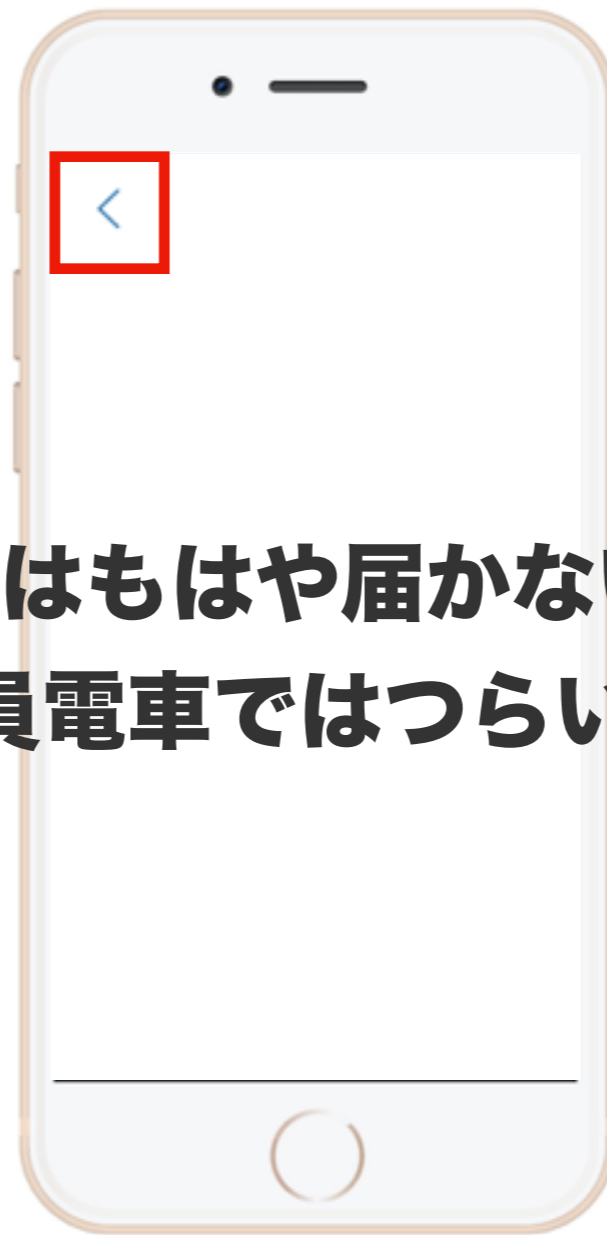


- ・ 一覧で読みたい記事を探して、個別記事へ遷移する。
- ・ 記事を読んだら、また一覧へ戻って、記事を探す。

iOSのガイドラインでは、左上に戻るボタン



iOSのガイドラインでは、左上に戻るボタン



**右手ではもはや届かない。  
(満員電車ではつらい)**

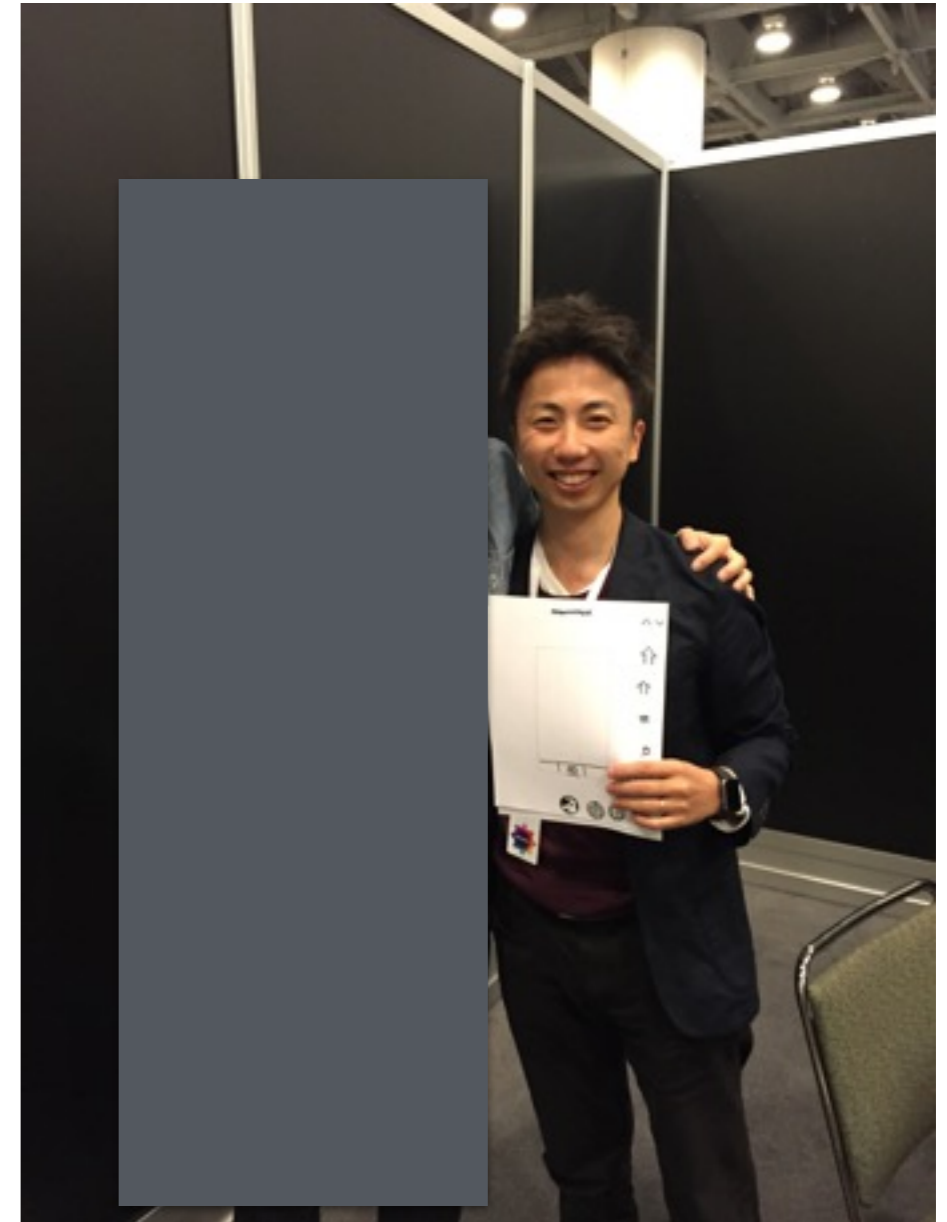
(Androidは、端末の戻るボタンがあるから話が早い。)

「iPhoneは画面大きくなってるけど、戻るボタンって右手で持ってたらず届かないっすよね？」

「ホームボタン2回タップすれば届くよ。」

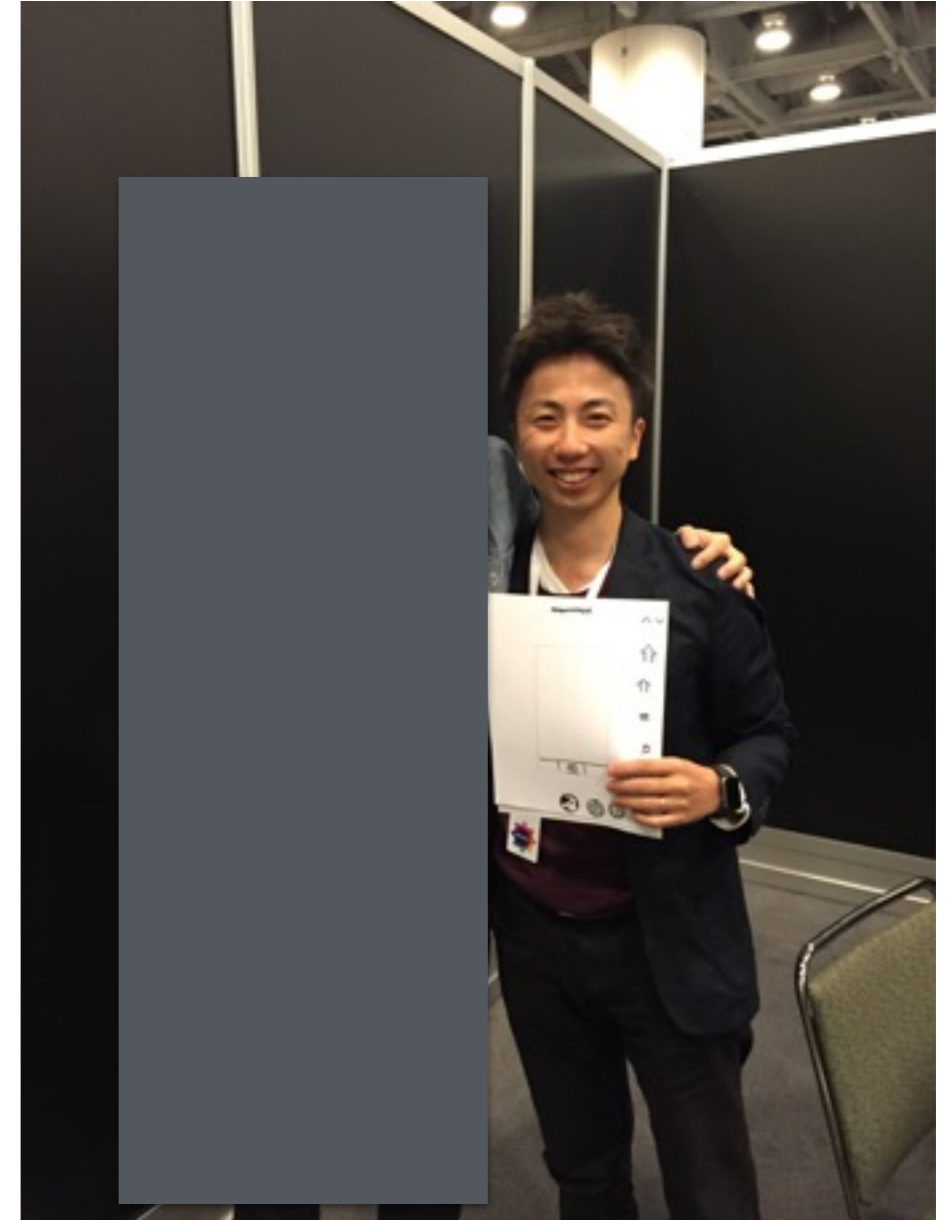
「・・・（それめっちゃ使いづらいよね）」

「あとはスワイプで戻るようにするといいよ」



「iPhone4SとiPhone6plusでは  
サイズが違いすぎるから、それぞれ  
UIを用意した方がいいの？」

「別に用意してもいいけど、Appleとしては  
もうiPhone6がスタンダードだよ。」



- スマートニュース  
グノシー  
LINEニュース  
などのニュースアプリで慣れ親しんだ操作。
- ユーザー調査でも、  
**Web刊をメインで読むユーザーには好評。**





## 記事一覧



## 個別記事



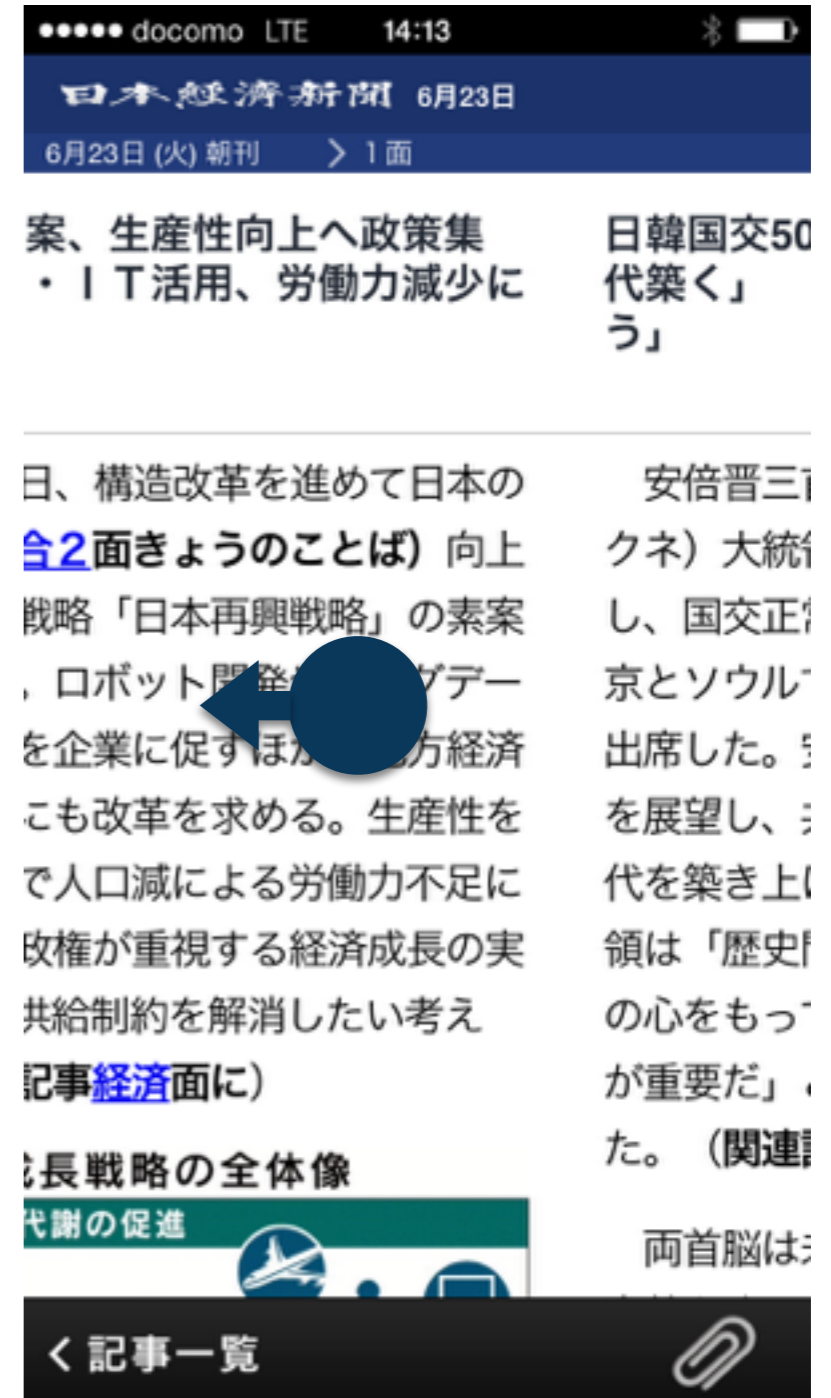
## 次の個別記事

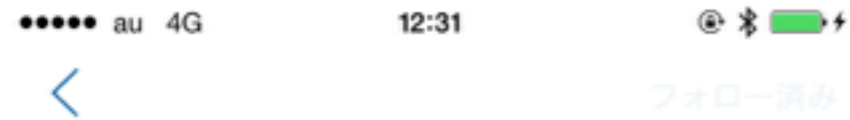


- ・ 個別記事に遷移したら、そのまま次の記事、次の記事と読み進める人も。
- ・ 記事を1本目からすべてチェックする。

# 旧アプリでは・・・

- ・ スワイプで次の記事・前の記事へ遷移できた。
- ・ でもスワイプは「一覧へ戻る」で使う・・・



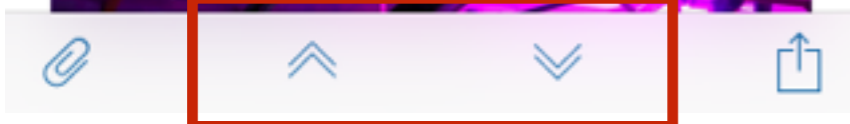
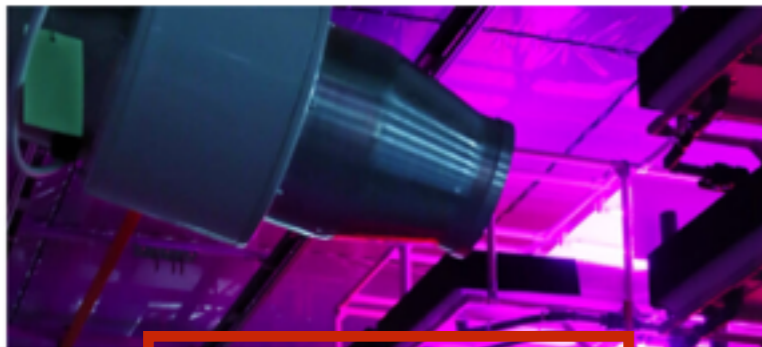


## パナソニック再興 デジカメからレタスへ

2015/6/20 6:30 産業新聞  
日経産業新聞 Editor's Choice

パナソニックが経営の主軸を大きく変える。2年連続の巨額赤字を招いた[デジタル家電](#)の構造改革にメドをつけ、ヒト・モノ・カネを「B to B（企業向け）事業」に注ぎ込む。2016年3月期を「成長に向けてスタートを切る年」と位置付ける津賀一宏社長はどんな将来像を描いているのか。パナソニック再興に向けた変化の波頭を追う。

### ■不良品ほぼゼロ



- ・ 前の記事・次の記事へ遷移するボタンを設置
- ・ 画面下中央の押しやすい位置に配置（右手でも左手でも押せる位置）



左右へ移動する一覧レイヤー



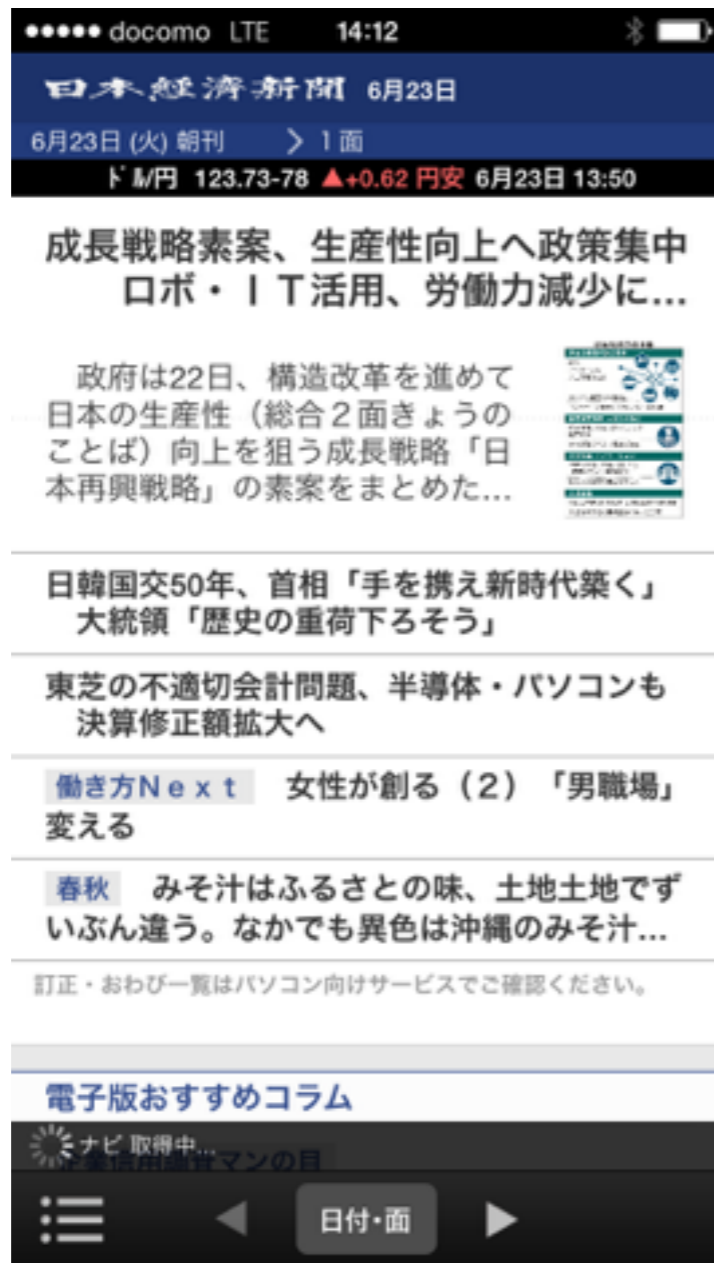
上下へ移動する一覧レイヤー

**リニューアルって胃が痛い**

リニューアルって胃が痛い

UIを変えることによる不満

## ガラッと変えると



## ガラッと変えると

「素晴らしいアップデートです」  
「まえと見違えるように使いやすくなった」  
「サクサクでいいアプリです」



「使いにくいし 見難いし」  
「パッと見は良いんだけど使いづらい…」  
「前の方が良かった」



**リニューアル後の対応が大事。**

**短期間でアップデートを繰り返し、ユーザーとの信頼関係を築く。**



## 《対応中のユーザーからの要望》

- ・フリック、スワイプなどの操作感の改善
- ・デフォルト起動画面の設定
- ・面をまたいでの記事間の移動
- ・Evernoteへの記事保存連携

などなど、絶賛内製開発中。

リニューアルって胃が痛い

**ユーザーの意見の本質を探る**

## 例) 「スワイプで次の記事へ移動したい」

もちろん、改修を検討するのだが、

**そもそもなぜ、新書2冊分の記事を順番に読むというニーズがあるのか。**

### 《仮説》

- ・ 新聞は見出しの大きさなどで、重要度（価値付け）を表現していて、読者はそれを見て読む記事をピックアップする。
- ・ テキストの記事一覧を見ただけでは「価値付け」がわからない。
- ・ だから、とりあえず順番に読む。
- ・ 「価値付け」がわかるようになればいい？

→紙面のイメージを見られるようにする。

**サクサク感を出すための実装**

サクサク感を出すための実装

一覧画面のスクロールがひっかかる

## 表示要素が多い

- (1) 見出しその1
- (2) 見出しその2
- (3) 記事の公開日時
- (4) 有料会員限定記事かどうかを示すアイコン
- (5) 記事内に動画があるかどうかを示すアイコン
- (6) 記事が所属する連載・コラム名
- (7) 記事本文のスニペット
- (8) サムネイル画像
- (9) サムネイル画像のクレジット

### FRBが隠す爪 迫る米利上げ、楽観の市場と溝

6/23 12:00   日経ヴェリタスセレクト

「ゴール（物価目標）に達する前に利上げを始めるべき」。4月、米連邦準備理事会（FRB）のパウエル理事はこう明言した。今月初…



Getty共同

UILabelなどを使わず、drawRectで直接描画したら速かった。

```
NKSLabel *titleLabel = NKSLabel.new;
self.titleLabel = titleLabel;
[self.contentView addSubview:self.titleLabel];
self.titleLabel.translatesAutoresizingMaskIntoConstraints = NO;
self.titleLabel.numberOfLines = 0;
self.titleLabel.font = [NKSSystemFont defaultBoldFontWithSize:[NKSSystemAppearance title1FontSize] + 2];

self.titleLabel.preferredMaxLayoutWidth = contentView.width - (largePadding * 2);
CGSize titleSize = [self.titleLabel sizeThatFits:CGSizeMake(self.titleLabel.preferredMaxLayoutWidth, CGFLOAT_MAX)];
self.addUiConstraints = [self.titleLabel mas_makeConstraints:^(MASConstraintMaker *make) {
    make.top.equalTo(self.thumbnailView.mas_bottom).with.offset(largePadding);
    make.left.equalTo(self.contentView.mas_left).with.offset(largePadding);
    make.bottom.lessThanOrEqualTo(self.title2Label.mas_top);
    make.right.equalTo(self.contentView.mas_right).with.offset(-largePadding);
    make.height.equalTo([NSNumber numberWithFloat:titleSize.height]);
}];

self.titleLabel.attributedString = [self.titleLabel stringApplyLargeSpaceInLines:headlineArticle.formattedTitle];
```

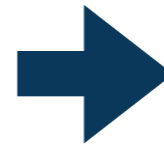


```
CGRect titleRect = [displayTitle boundingRect:remainRect
                    options:NSStringDrawingUsesLineFragmentOrigin | NSStringDrawingTruncatesLastVisibleLine
                    context:nil];

[displayTitle drawWithRect:titleRect
                options:NSStringDrawingUsesLineFragmentOrigin | NSStringDrawingTruncatesLastVisibleLine
                context:nil];
```

# 一覧画面のスクロールがひっかかる

## Before



## After





# 一覧画面のスクロールがひっかかる



## Before



## After



Appleの審査手続き中

サクサク感を出すための実装

**データ取得後の保存処理に時間がかかる**

iPhone4などの低スペック端末で顕著

- ・ 記事数が多い（なんせ新書2冊分）
- ・ 記事に紐づくデータも多い

## CoreDataをやめ、Realmを組み込む。

CoreDataは関連をつけたオブジェクトをまとめて500件ほど保存すると、パフォーマンスが急激に落ちる。

Realmは関連をつけても高速で動作する。

**Realm** is a mobile database

Realm is a replacement for SQLite & Core Data.  
It can save you thousands of lines of code & weeks of work,  
and lets you craft amazing new user experiences.

Download Realm

Objective-C Swift Android

Enter your email to join our community newsletter

Sign up for our community newsletter to hear about Realm tutorials, events, tips & more!

GROUPON Expensify hipmunk Pinterest  
zipcar RITE AID SAP 3M

<https://realm.io/>

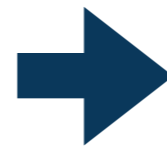
# データ取得後の保存処理に時間がかかる

iPhone4/iOS7である日の朝刊を取得

**Before**

**After**

00:32.16  
00:32.16



00:04.68  
00:04.68

# データ取得後の保存処理に時間がかかる

iPhone4/iOS7である日の朝刊を取得

Before

After

00:32.16

00:04.68

00:32.16

**Appleの審査手続き中**

00:04.68

**まとめ**

- **実際の使われ方を徹底的に調べる。**
- **仮説は立てるが、最後はおもいきってやってみるしかない。**
- **リニューアル後の高速PDCAが勝負。**
- **ユーザーの意見をきちんと咀嚼する。**

**宣伝させてください。**



- **少しぐらい障害があった方が燃える方**
- **フロントエンド以外ももちろんOK**
- **とりあえず職場のぞいてみたい、でもOK。**

**下記まで気軽に連絡ください。**  
**dg\_lab@nex.nikkei.co.jp**

**ありがとうございました。**