

EXHIBIT 165

UNREDACTED VERSION OF DOCUMENT SOUGHT TO BE LODGED UNDER SEAL

From: Douglas Purdy </O=THEFACEBOOK/OU=EXCHANGE ADMINISTRATIVE GROUP (FYDIBOHF23SPDLT)/CN=RECIPIENTS/CN=DOUGLAS PURDY>
Sent: Thursday, April 11, 2013 12:01 AM
To: Mike Vernal
Subject: Re: Platform Rules of the Road (early draft)

I am looking for the last thing from mark, that is not where I think we landed (but I think we are both remembering what we want. I let this lay too long after Charles left, my fault).

On Apr 10, 2013, at 11:59 PM, "Mike Vernal" <vernal@fb.com> wrote:

Sorry, I switched names when writing this email.

"Social reciprocity" is what we called "Total reciprocity" in our conversations w/ Mark. Does that help?

From: Douglas Purdy <dmp@fb.com>
Date: Wednesday, April 10, 2013 11:54 PM
To: k a <vernal@fb.com>
Subject: Re: Platform Rules of the Road (early draft)

1. Read the other mail. I think it has the Dan level stuff.
2. The social reciprocity thing is complete new to me. We decided to do app friends in the meeting with mark. I modified that for canvas in my other note. I think that is really complex. I buy the categorical model and that is what I meant (new should be new or edit) and will make clearer.

On Apr 10, 2013, at 11:46 PM, "Mike Vernal" <vernal@fb.com> wrote:

Ok, sorry - will read/respond to the other email. That said, this is the level of detail that I think those folks want.

Re: reciprocity - can you provide some more detailed feedback here? This is the thing I feel most strongly about. If you think simplifying it's important, I could see saying:

- If you access any user data or the friend graph, you have to implement full reciprocity
- If you access the friend graph, you have to implement full reciprocity (you can always access user data)

The world I (strongly) don't want to be in is the "if you want to use our APIs, you have to publish data back to us (even if it doesn't make sense for your app)".

-mike

From: Douglas Purdy <dmp@fb.com>
Date: Wednesday, April 10, 2013 11:42 PM
To: k a <vernal@fb.com>
Subject: Re: Platform Rules of the Road (early draft)

First, you are reading things in the wrong order and this email only makes sense in the context of the first mail.

Second, I don't want to get to this level of detail with Dan and Justin. I have 2 PMs working on this issue and I want them to help determine the right thing to do based on core principles we construct and really usage data (we make dumb decisions in the room with a bunch of directors and no data).

In terms of reciprocity, I work here and I don't even know what you said (hyperbole), so I think the model you outline is really hard to follow so we should discuss.

On Apr 10, 2013, at 11:34 PM, "Mike Vernal" <vernal@fb.com> wrote:

Thanks for writing this up.

I have a lot of thoughts. I'll try to keep this succinct.

Reciprocity - as the inventor of reciprocity, I think the vision I had for this has gotten distorted a little bit. Here's where _I_ think we ended-up in November:

- User Data - users can always bring their data to third-party apps (the photos they uploaded, the likes they made, etc.)
- Friend Data - we are removing the ability to access non-PAI for friends
- Categorical Reciprocity - if an app developer reads a certain type of user data (e.g., photos), and the user can also create new data of this kind in your app (new photos), the developer must:
 - Offer a prominent way to share this back to Facebook
 - Implement the Action Importer API so that users can import this type of data into Facebook
- Social Reciprocity - if an app developer reads the friend graph, then they must also:
 - Offer a prominent way to share all social actions back to Facebook
 - Implement the Action Importer API so that users can import their actions into Facebook

The really important difference here is that I (strongly) think we should continue to support apps that offer compelling, social experiences, even if those apps don't have any natural sharing back to Facebook. If an app creates data or creates social actions, I think users need to be able to bring that data back to Facebook, but if an app doesn't create any data or social actions, we shouldn't ban that app. I feel really strongly about this, but I think there's either disagreement or confusion on this point (you wrote this below, I've heard Justin say this, I've heard Dan say this, so we may need to discuss).

Strategic Competitors - I think your wording/framing below is too nuanced. I think this is basically "there is a very small number of apps that have been flagged as strategic competitors. All of their uses of our platform (ads, identity, and app services) require prior review/approval from Mark."

API Classes - you are correct in channeling me. I thought about this for a little while, and here's a suggested framing:

- Read
 - Login (uid, name, profile pic, small # of core fields) - anyone can get this. No a priori review.
 - User Data - requires user_* permissions. Ability to ask for those user_* permissions requires unified review.
 - Friend List - Requires unified review. If you access the friend list, you must conform with Social Reciprocity, as defined above.
 - *Friend Data - we're removing this (removing friend_* permissions)*
 - *Core Facebook Features (News Feed API, Inbox API, Full Search API, etc.) - requires unified review. Generally only available with a business deal, generally limited to Facebook replacement apps.*
- Write
 - Share Dialog - anyone can use
 - Open Graph (defining actions, using the API) - requires unified review
 - *Other Write/Management APIs (events, groups, etc.) - requires unified review. Generally requires a business deal, only available to Facebook replacement apps.*
- Distribution
 - Bookmarks - limited to canvas apps + mobile games. Requires unified review.
 - Requests - limited to canvas apps + mobile games. Requires unified review.
 - Notifications - limited to canvas apps + mobile games. Requires unified review.
 - App Center + Search - limited to apps that have gone through unified review.
 - Messaging (Invitations) - will require unified review. Available to anyone who abides by our rules (does not spam the channel).

Your Open Questions:

- Non-Writeable Fields - I think we should continue to expose birthday, hometown, etc. We can add write APIs whenever it makes sense, but we shouldn't remove these fields because we don't have write APIs today.
- Canvas apps - I think they're just subject to the same rules as above. I don't think this is an issue if you use my definition of reciprocity.

- Facebook Payments - I think I'd let this stand on its own legs and not couple it to login.

For the meeting tomorrow, I (personally) think the two biggest issues to probably get agreement on is (a) the crisp definition of reciprocity and (b) something like the API classes thing I mention above. Your call - not trying to be prescriptive, but trying to make sure we have an effective conversation.

From: Douglas Purdy <dmp@fb.com>
Date: Wednesday, April 10, 2013 8:27 PM
To: k a <vernal@fb.com>
Subject: FW: Platform Rules of the Road (early draft)

I'll channel you and say that this is too high-level still. I agree. We are doing a API by API review to determine the review rules, which we are going to bucket up into a doc and in the tool. The goal of the below is to get everyone on the same broad page. Feedback welcomed.

From: Douglas Purdy <dmp@fb.com>
Date: Wednesday, April 10, 2013 8:25 PM
To: Constantin Koumouzelis <constantin@fb.com>, Marie Hagman <marieh@fb.com>, Zhen Fang <zhen@fb.com>, Eddie O'Neil <ekoneil@fb.com>, George Lee <george@fb.com>
Subject: Platform Rules of the Road (early draft)

[Below is an broad summary of our app review guidelines. I really want to encode most of this in the devops review tool. Could you folks look at this and comment, flame, whatever?]

As we work towards implementing the decisions that we made last year, which are now known as Platform 3.0, we need a common framework by which we can make decisions about what types of app to give access to Platform. This framework must address three key questions: what are the broad principles of our platform, how do these manifest in our products/policies and how do we communicate this to developers? This document answers these questions, constituting the Platform "rules of the road".

Principles

The fundamental principle that governs Platform usage is a simple concept: reciprocity. Reciprocity involves a equalible value exchange between a 3rd party developer and Facebook. This value exchange involves one of the following from developers: high-quality experiences that FB users can use to tell great stories to their friends and family on FB and/or monetary value in the form of

revenue sharing or direct payment. In return, Facebook offers a developers access to our Platform.

When considering the implications of reciprocity it is important to note that a second order principle quickly emerges: competitive access. There are a small number of developers whom no amount of sharing to FB or monetary value can justify giving them access to Platform. These developers do not want to participate in the ecosystem we have created, but rather build their own ecosystem at the expense of our users, other developers and, of course, us. That is something that we will not allow.

Platform Services

In order to outline how the above principles manifest in our products/policies, we need to identify the various parts of Platform. This is required because we have a disjoint set of product and policy constraints that apply to each of these different areas:

App services: these are paid generic services (storage, compute, etc.) that apps may use to build the core foundation of their app. At present, we do not have an offering in this space, but we are close to closing an acquisition that adds these services. As such, in order to be complete and future-proof we will outline the rules associated with these types of services.

Ads services: these are paid promotional services that enable developers to drive awareness and installations of their apps using News Feed and other paid channels. We have always had an advertising business the developers could leverage, but this is increasingly an area of focus for us with the transition to mobile.

Identity services: these are the traditional identity/social services that we have provided to developers since 2007. These services enable developers to build personalized app experiences for people and enable these people to share aspects of those experiences back to Facebook. [todo: payments is here]

Application

The following outlines the application of the above principles to the various kinds of platform services.

Strategic competitors: We maintain a small list of strategic competitors that Mark personally reviewed. Apps produced by the companies on this list are subject to a number of restrictions outlined below. Any usage beyond that specified is not permitted without Mark level sign-off.

Ad services: All developers, save strategic competitors (above), may use our ads services. The reciprocity for these services is clear: money in exchange for new or re-

engaged users. In terms of oversight/policy enforcement, we follow the standard ads creative review process.

App services: All developers, save strategic competitors (above), may use our app services. The reciprocity for these services is clear: money in exchange for CPU, data storage and network bandwidth. In terms of oversight/policy enforcement, we will reactive handle any strategic competitors that we discover using these services.

Identity services: this set of services is the subject of much of the rules of the road. This is due to the fact that we have a variety of mechanisms for value exchange.

All developers, **including strategic competitors (above)**, may use the Login and Social Plugin features available within identity services. We permit this because we wish to see our core login service and basic sharing services used by users in any app, creating an equitable relationship with the all, including competitive, developers. To this end, we make these features available to developers with out app review.

The use of identity services, beyond Login and Social Plugins, is subject to app review. We review the apps usage of our APIs to determine if they are adhering to our principles. In particular, we look at the quality of the app's user experience and if there existing some equitable value exchange.

During app review, we determine the quality of the app by using the app and comparing the experience to our quality guidelines [link]. Apps that do not meet our quality bar are rejected.

During app review, we examine the APIs that the app uses in order to determine what the appropriate level of reciprocity. For apps that do not use our payments service (which is direct form of equitable value exchange), the guideline for this review is "take data, give data". For example, if the app accesses the user's photos, the app must offer a prominent mechanism for the user to share new or modified photos back to Facebook. The review tool is built to help with this assessment in that for every read API used by the app, we flag if the app is also writing this data back to Facebook. Using this tool, as well as an examination of the user experience itself, we can determine if the app is reciprocating. If they are not, the app is a "data leach" and will be rejected.

Open Issues

There are a number of different fields like birthday, hometown, etc. that apps can request and there is no way for them to write back anything. We can do a couple of things: create an API to set this info (maybe not a bad idea for identity growth?), limit these data

fields to just canvas apps (the value exchange is time on site and maybe payments), pull these fields, something else?

How do we think about the baseline level of value exchange of canvas apps due to time on site? Is that enough to forgo the "take data, give data" mandate for non-payment games/apps?

If you are offering real/world goods for sale on your web site or mobile app, in order to use identity services, you must use Facebook Wallet (Payments 3.0)?