

## EXHIBIT 36

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, August 30, 2012 10:17 AM  
**To:** Sam Lessin  
**Cc:** Mike Vernal  
**Subject:** Re: Platform Business Model Framing

I am on vacation this week (still), but happy to help.

That said, in wonder if you should just take the pen on the deck moving forward so I can have a day off and not be the bottle neck?

I can send you what I have so far in an hour when I get back to a computer.

On Aug 30, 2012, at 10:11 AM, "Sam Lessin" <[sl@fb.com](mailto:sl@fb.com)> wrote:

1. Sorry I wasn't clearer about my plans / being out. That is my bad guys / didn't mean to leave this hanging (obviously super important)
2. Doug, really glad you are framing this up / working on this ...
3. Here are my notes on where things are from Friday, I think a lot of it is covered below but I just want to make sure we are in sync here, are presenting all the hard questions for this offsite

(A) How the platform / our APIs would work if we could start over / the base rules for all businesses using Facebook.

- There are two basic sides to platform that function differently
  - There is a 'write' side, whereby businesses can get permission to write to the graph on behalf of users
  - There is a 'read' side, whereby businesses can get permission to read from the graph on behalf of users
- On the 'write' side:
  - The value proposition: you can write to our system both messages 'on behalf' of users (think explicit posts and timeline boxes), and messages on your own behalf (think page posts), to drive growth and re-engagement. We give you a natural amount of distribution for free / to make our user experience best, and we charge for everything else.
    - Applications can write whatever they want to the graph on behalf of users (with the user's permission in most cases)
    - Distribution
      - All content written by applications gets the natural amount of newsfeed distribution based on the NF algorithms for maximizing engagement & user happiness
      - *Any / All applications can pay to up-rank themselves in feed if they want more traffic*

- *Any / All applications can pay to get into premium channels (invites, inbox, etc)*
  - Information
    - If you write structured data about a user in a way that other businesses want to use for targeting \*we will pay you\* (mechanism TBD) and you can set certain limitations on how the data can be used / by what parties.
- On the 'read' side:
  - The value proposition: you can read from our system. Users can always give you the information they have given us directly in order to help you customize your service / provide better service. We, Facebook, have more information based on our own derivation, as well as aggregate information about the connections/relationships of a user which we will give you if your service meets certain requirements and you are willing to pay us.
    - Applications can use Facebook for 'login' to allow users of their application to not need to remember another password, etc.
    - User-Data
      - Applications can request a user give them 'their data' in order to provide the user with a more custom experience (*NB: the 'user's data' is explicitly the data which the user has entered themselves into Facebook - e.g. Name, profile photo, hometown, etc.- .... It is not tagged content, etc.*). This effectively resolves to an improved 'registration plugin'
      - Applications cannot request users give them 'contact information' including UID, email, phone; however, we do need to allow applications some way of reaching their users in a stable manner using Facebook (e.g. At a minimum spam folder in inbox, or a proxy email, or something)
      - Applications cannot request users give them 'friends' of the user (which isn't purely the user's data because it requires confirmation, etc.) ; however, (1) we allow a given app to get 'friends of this user who are also using this application' and (2) we do provide a paid 'invite' channel whereby applications can ask a user to invite more of their relationships to the application.
      - Applications cannot request information about a user's 'friends' at all, nor can they request feed, etc.
    - Our Data
      - Facebook has information about users which can be helpful to applications, and which we provide to applications when we deem appropriate, this information includes:
        - Aggregate data about the tastes, properties, etc. of a user's friends (things this user's friends like, places they live, etc.)
        - Derived data about a user / facebook's data/opinion of a user (probable location, account trust score, account age, etc.)
        - (*Data provided by third parties — information which third parties have contributed to the graph on behalf of a user*)
      - *We allow a limited number of calls for free to Any / All applications for this data (and always free calls for the admins of the app)*

- *We charge everyone on a per-call basis for this information. We have a standard rate card, but require a deal and price based on understanding of / commitment of app re: use*
- *We do not allow things which are at all competitive to 'buy' this data from us*
- Mechanics:
  - From a mechanics perspective, the APIs function much as they do today; however, if we could start over / the way platform would ideally work, we never give applications UIDs, we only give out APP-specific hashed IDs. While more complicated, this provides a significant technical hurdle for applications trying to 'suck out' the graph and makes sure that apps using our platform are committed to our distribution channels.

(B) Extending the platform to make it more valuable for engagement by anonymous writes and targeting. **The question here is what is the full opportunity / how big a deal or valuable is it for these functions to exist.**

- By default / in our base case applications can only write data 'on behalf' of users, or on their own behalf, but they can't write data 'about' users without the user's express consent via GDP, etc.
- In an 'extended' world, we would do the following:
  - Allow applications to 'target' on Facebook any user that has interacted with them in a broader set of ways (visited page, opened photo, seen post in feed, etc.) -- not underway
  - Allow applications to pass us a email address and get back a hashed-ID for a user which they can then use to communicate with / advertise to / message user -- underway in basic form
  - Allow applications to drop pixels on anything they want on the web / etc. which fire back open graph edges against the hashed-ID (visited page, took action, etc.) which the app can set as 'only me' (aka, completely private to the application) or 'me + user' (which includes the user in the privacy of the post) -- not underway
  - Allow applications to write any other data they want directly in against a user's hashed-ID for their own targeting use AND which they can allow other businesses to target against given certain terms / black list & rev-share, etc. -- not underway

(C) How to properly 'sell' information so that we are happy with the results of doing this net of the cost of change.

- How much can we make short and long-term selling data
- What is the developer thrash / cost of moving from free to paid for new or existing APIs
- We know we can't efficiently price it... how do we price it at all, etc.

---

**From:** Douglas Purdy <dmp@fb.com>  
**Date:** Wednesday, August 29, 2012 12:47 AM  
**To:** Mike Vernal <vernal@fb.com>  
**Cc:** Sam Lessin <sl@fb.com>  
**Subject:** Re: Platform Business Model Framing

working on this.

going to see the attached to frame things up.

On Aug 28, 2012, at 6:15 PM, Mike Vernal <vernal@fb.com> wrote:

Doug - as context, we're having an mteam offsite on Tue + Wed of next week to talk about three-year-plan stuff, and one of the discussions we're going to have is around the Open Graph business model.

Sam is at burning man (not sure when he gets back), and we left it a little ambiguous about who was pulling together what, so I'd like to at least get started pulling together a deck that we can use to frame the conversation. Can you ask the PMs to pull together a few slides?

Ideally, I think we want to cover:

Read APIs:

- What read APIs are free\*, and cost-recovery charging for free APIs
- Premium read APIs
- Rules around using read APIs

Write APIs:

- Free write APIs (specifically, the Open Graph APIs)
- Premium distribution APIs
- Monetizing this data (sponsored stories, contextual ads on FB, contextual ads off FB)

Developer Rules:

- Anyone can build for free, have to pay a developer fee to launch
- Goes through app review process, we review Facebook integration (edges it publishes) and app assets, goes into app center
- We re-charge yearly, re-review periodically

CRM / Data Exchange Scenarios:

- "Anonymous" write APIs - the CRM/data broker scenario that Sam is passionate about
- Monetizing this data

Not sure of the best framing (and I'm a little feverish right now), but it would be good to at least start pulling this together. Sam + I spent a couple of hours talking about this on Friday evening, and he had a framework for the read API stuff and the CRM stuff he was going to write-up. Otherwise, I think you/PMs have most/all the context.

For the read stuff, I think the basic framing was:

- We move all IDs from UIDs to per-app/hashed IDs in the system
- You can always bring your data to an app (your statuses, your photos, your events, etc.)

- You can bring your friends that are also using the app to the app; ideally you can't bring non-app-user-friends to the app (you have to go through us to invite them)
- We get rid of the ability for you to take your friends' data to an app
- We have a set of premium APIs like coefficient, SI, etc. where you can pay us per-user to get this data. We set the price.
- The CRM stuff I think is the idea that others can also write this data into the system and attach a charge for other people to use this data.

I think Sam is probably the right person to write-up the CRM stuff, but the rest I think the PMs have the context on.

-mike