

EXHIBIT 39

UNREDACTED VERSION OF DOCUMENT SOUGHT TO BE LODGED UNDER SEAL

OG as a business

8/2012

The core value exchange of Open Graph is centered on distribution and data...

	Value to Developers	Value to Users	Value to Facebook
Distribution Apps 'write' to facebook	-Users & Traffic (acquisition & re-engagement)	-Self expression within facebook	-User engagement - 'Boostable' content which drives spend - Targetable data which drives spend
Data Apps 'read' from facebook	- Personalization & inviting friends - Option value / monetization in the future	- Better user experiences in apps	- Presence across the web - Data Portability

We face a series of issues with the value model...

Distribution

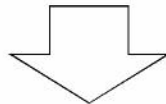
Apps 'write' to facebook

- User engagement in many cases (not all) negative & without objective function we thrash developers on distribution
- Developers are not advertising / boosting
- Data so far not useful for targeting & face conflict with developers over 'value' of their data

Data

Apps 'read' from facebook

- We are enabling competitors / do not have rules that make us truly comfortable with path, pinterest, etc.
- Currently unclear value to Facebook
- No natural market dynamics, hard to correctly price
- Undermining user trust when developers have more data than they need



**OG lacks a clear value exchange and currently drives
<\$10M in revenue**

The Current Opportunity Matrix:

		E		D		\$		Note & Risks
Area	Product	Now	Need	Now	Need	2013	18RR	
Access	Annual Developer Fee	0	5	0	1	\$5	\$7	Eng is mostly for support & QA. We are all agreed this is good housekeeping but not a big business.
Access	Device Integration Partners	0	3	0	0	\$2	\$2	Eng is mostly for support & QA. This could be an issue with integration partners / reverse of thinking.
Read	API usage fees on current	0	4	0	1	\$7	\$9	Majority of value at the high end / could be higher if we charge big developers more. Good housekeeping.
Read	Premium read APIs (coefficient, SI, u-to-u)	1	9	0	1	\$3	\$140	18-mo run-rate mostly top down with market share assumption, high risk
Write	Invitations Channel (mobile)	1	3	0	<1	\$14	\$28	60m outbound invite clicks -> 6m installs / mo. Assumes ~\$1 / install + lift.
Write	Notifications / Paid Messaging	0	3	0	<1	\$15	\$30	Re-engagement charged per-click at 0.25% of install.... Competing with push notifications. <i>Tightly tied to other paid messaging.</i>
Write	'Apps you may like' (Neko)	3	7	0	1	\$200	\$260+	Based on market share analysis. Relatively high confidence in the opportunity.
Write	App Sponsored Stories	0	4	0	1	\$ -	\$ -	Overlaps with ads / general sponsored stories storyline. There is an opportunity with developers of TBD size.
Write	Placement in App Center	0	1	<1	<1	\$18	\$18	No meaningful mobile traffic / tiny opportunity now. On web is canvas focused / legacy business so no growth.
Write	Sponsored Search Results	0	0	0	0	\$30	\$30	Mobile outbound from search is tiny currently... Making this opportunity small in next 18 mo.
Write	Contextual Ads on FB (amazon-pinterest case)	0	10	0	2	\$40	\$90	Potential serious issues with partners in terms of use of / direct monetization of their data. Also user concerns. Inventory is a problem.
Write	Contextual Ads on 3 rd Party Sites (ad-network is a pre-requisite)	0	10	0	1	\$4	\$7	We keep a far smaller amount of the revenue here which is an issue when we have excess inventory. Also won't scale until inventory-constrained on FB.
CRM	Ad-Targeting based on own data	X	X	X	X	\$-	\$-	Heavy overlap with Ads team... this is a minor functional extension if we decide to build a real CRM product
CRM	Third-party data exchange	X	X	X	X	\$-	\$-	Heavy overlap with Ads team... not suggested until we have established

Access / Facebook Developer Program

- Program
 - Development is free (so is social plugin use)
 - Apps must be reviewed and pay a fee to go “live” to users
 - The fee covers the calendar year, re-charge on Jan. 1
 - We provide a small number of official support incidents (devsupport@)
 - Additional support can be purchased
- Benefits
 - Eliminates a class of spam/fraud apps on Platform
 - Deterministic support channel that lasts after launch
 - Sets the “paid” bit with developers
- Revenue
 - ~\$5-7M

Access / Device Integration Program

- Program
 - 16+ partners that create FB experiences on devices
 - Provides paid, dedicated support for these partners
 - Provides testing to ensure that we don't break them
 - We will restrict some APIs (like stream.get) to these partners
- Revenue
 - ~\$2M

Basic Read APIs

- After the developer has paid, they can request access to:
 - A user's data (basic info, photos, etc.)
 - A user's friend data, if the friend already uses the app
 - No other friend data is available (no friends.get) – *discussion topic*
 - Potentially for extra fees allow custom GDP to help conversion
- Once granted, we charge a cost recovery fee for data access to these APIs
 - CPU time: \$.00007/s or \$0.25/h
- Revenue
 - ~\$7-9M

Premium Read APIs

- Provides our 'derivative' user data to apps
- Current service explorations underway
 - User Trust Service: A trust score for a given user
 - Recommendation Service: What OG objects a user may like (music, books, etc.)
 - Co-efficient Service: How strong the relationship is between two users
- Revenue
 - ramping to order of ~\$140M

Write (A series of opportunities overlapping with Ads)

- Program
 - All apps can write OG and stream.publish, etc. to user's timelines for free
 - Apps get rational distribution in feed - no boosting/inflation
 - This implies throttling back from where we are in many places today
 - Apps can pay for distribution via a variety of channels
 - Invitation Channel (14M/28M)
 - Notifications / Paid Messaging (15M/30M)
 - Apps you may like (Neko) (200M/260M)
 - Sponsored Stories (-M/-M)
 - Placement in App Center (18M/18M)
 - Sponsored Search Results (30M/30M)
 - Ideally apps bid on install value and reengagement value, and we optimize across these channels
- Revenue
 - ~\$300-400M, primarily driven by Neko, overlapping with ads

Become a Marketplace

- Current OG Sponsored Story Problem
 - Our best developers can generate good OG actions, but don't have money to spend on ads
 - Our best advertisers have a lot of money to spend, but can't generate OG actions
 - Many early developers have a product, but lack a business model / revenue
- Solution
 - Enable advertisers to attach sponsored actions to OG content
 - Mike reads a book on Goodreads. – Buy on Amazon
 - Sam rated a movie on Rotten Tomatoes. – Watch on Netflix
 - Sponsoring boosts story and attaches a prominent UI action link
 - Rev-share with developer to align incentives and provide revenue
- Longer Term...
 - When we become inventory constrained, enable developers to have sponsored stories / action links on their site, too.

These opportunities basically boil down to...

Basic Fees

- A fee for launching applications will not generate meaningful revenue, but it will help us focus, give support, and cut down on spam / bad app interactions

Distribution

Apps 'write' to facebook

- Stop artificially 'boosting' distribution for free: developers can buy sponsored stories if they want to drive traffic, but should be on equal footing with other businesses
- Open up new stable paid distribution channels beyond newsfeed e.g. invitations, notifications, messages
- Give developers the ability to target non-connected users via various existing channels (email import, RTB, etc.), and beyond
- Launch ad-network on mobile (and eventually desktop) to help applications monetize their apps / give us a way to get more value from platform

Data

Apps 'read' from facebook

- Charge developers for use of APIs (on a cost-recovery+ pricing model), starting with a graduated pay scale charging larger developers more
- Introduce new 'premium' paid APIs that are particularly valuable to certain industries (e.g. authenticity data) & develop partnerships / rev-share for value-added third parties
- Require data reciprocity as a principle and a policy; for top partners / competitors require a deal where we can let users import their data in return for read API access

Marketplace

- Monetize partner data + give them a cut of the value.

To make our channels effective and clarify our value proposition, we should consider certain other changes vs. cost of 'developer thrash'

Stop giving out 'contact information'

- We have no interest / incentive in apps having the email, phone of users. Taking this away makes our ability to proxy communication to users more valuable.

Shift away from UID-based APIs

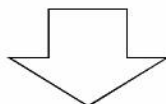
- Identity is our main point of value. By giving out universal identifiers for things in the graph, we make it easy for others to both remove / recombine parts of the graph AND withhold information from the graph / trade it outside of our market.

Modify friends.get & move away from non-registration data

- Developers use friends.get to grow their apps, while we simultaneously attempt to sell the same / compete with ourselves. Apps should be able to (1) target all of the friends of a user on FB (2) get the friends of a user also using their app
- More broadly, we should deprecate users taking the data of others to apps & be selling services to help applications efficiently divine the likes and interests of a user based on those of friends, etc. rather than encouraging crawling and sucking in the graph

Over time, we should also consider anonymous writes & targeting options to extend the value of platform...

- 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.
- We would do the following (*some of which are underway across the org*):
 - **On-Site Targeting:** 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.)
 - **Email Matching:** Allow applications to pass us an email address and get back a hashed-ID for a user which they can then use to communicate with / advertise to / message user
 - **More Robust 'Conversion' Tracking:** 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.)
 - **Data-Exchange / CRM :** Allow applications to write any other data they want directly in against any user's hashed-ID without GDP for their own targeting use that of others (with a rev-share)



These are hard to size for developers, and have other issues, unclear if we should pursue now or in the future