

Dazu wird die ID der Applikation über das *fb:app_id*-Metatag an das entsprechende Objekt gebunden.

- `<meta property="fb:app_id"`
- `content="[Applikations-ID]" />`

Sobald ein Objekt an eine Applikation gebunden ist, kann diese entsprechende Stream-Meldungen für das Objekt vornehmen und auch die Statistik Daten über die Graph API abrufen (vgl. Kapitel 8). Die eindeutige URL der Applikation (per *og:url*-Metatag spezifiziert) stellt hierbei die eindeutige ID des Objekts dar, womit dieses auch über die API abgerufen werden kann.

- `https://graph.facebook.com/?`
- `ids=http://www.databecker.de/buch/facebook`

Optimaler Einsatz des Open Graph Protokolls

Der Vorgang zum „Mögen“ eines Objekts sollte für einen Benutzer so einfach wie möglich gehalten werden. Die Übersichtlichkeit und optimale Position der Schaltfläche in direktem Bezug zum Objekt spielt eine entscheidene Rolle. Zudem sollten nur reelle Objekte als solche definiert werden – kein Benutzer wird die Verlinkung seines Profils mit News-Artikeln oder ähnlichen Inhalten positiv begrüßen. Zu guter Letzt sollte man sich bei der Veröffentlichung von Inhalten auf das Rahmenthema begrenzen. Findet ein Benutzer beispielsweise ein Buch gut, möchte er nicht ständig über generelle Informationen des Verlags informiert werden – dafür kann er den entsprechenden Verlag als „gemocht“ markieren und passende Aktualisierungen erhalten.

4.2 Schneller Start mit sozialen Widgets

Widget

Ein Widget ist eine Miniapplikation, die über eine gemeinsame Schnittstelle in eine andere Applikation integriert werden kann. Es kann prinzipiell nicht eigenständig betrieben werden, da es als Gast auf die Umgebung einer Wirtsapplikation angewiesen ist.

Um eine Seite intensiver in den Open Graph integrieren zu können, gibt es eine Reihe von entsprechenden Widgets, die wenig bis kein technisches Know-how voraussetzen. Jedes Widget wird durch einen Quellcode-Einzeiler repräsentiert – die Funktionalität befindet sich an zentraler Stelle auf der Seite von Facebook.

Im Detail handelt es sich bei jedem Widget entweder um einen HTML-Inline-Frame (IFrame), der den entsprechenden Inhalt direkt von Facebook lädt, oder um ein XFBML-Element.

```
■ <iframe src="http://www.facebook.com/plugins/..."
■     scrolling="no"
■     frameborder="0"
■     allowTransparency="true"
5     style="border:none; overflow:hidden;
■         width:300px; height:300px">
■ </iframe>
■
■ <!-- oder: -->
10
■ <fb:... />
```

Bei Verwendung dieser Widgets ist auch keine eigene Authentifizierung des Nutzers notwendig. Facebook verwendet eigene Cookies, um festzustellen, um welchen Benutzer es sich handelt und ob dieser eingeloggt ist. Sollte ein Benutzer nicht in Facebook eingeloggt sein, werden dennoch öffentliche Daten angezeigt und dem Benutzer wird eine entsprechende Möglichkeit geboten, sich einzuloggen.

Bei der Integration eines Widgets über einen Inline-Frame existiert für die entsprechende Webseite oder Applikation, die ein solches Widget nutzt, keine Möglichkeit, selbst Benutzerdaten über diesen Weg zu nutzen, geschweige denn zu speichern. Die gesamte Funktionalität liegt auf der Seite von Facebook.

Meistens bietet die Integration entsprechender Widgets per XFBML allerdings diverse Vorteile in Form von zusätzlicher Funktionalität. Bei der „Gefällt mir“-Schaltfläche beispielsweise kann ein Benutzer im Falle der Integration per XFBML zusätzlich das Objekt kommentieren. Die Nutzung von XFBML setzt allerdings auch die Integration der JavaScript-Bibliothek voraus (vgl. Kapitel 3.5).

Innerhalb einer Webseite oder Applikation können außerdem ohne Einschränkung mehrere Widgets kombiniert werden. Der zur Integration benötigte Quellcode kann mit einem Assistenten (<http://developers.facebook.com/plugins>) erstellt werden.

Jedes Widget bietet außerdem zahlreiche visuelle wie auch funktionelle Einstellungsmöglichkeiten. Der Umfang der vorhandenen Widgets wird stetig weiterentwickelt, einen genauen Überblick liefert stets der Widget-Assistent.

XFBML

Die eigentliche Funktionalität der Widgets wird von XFBML (**Cross Facebook Markup Language**) übernommen. XFBML unterscheidet sich dabei marginal vom einfachen FBML. Hier existieren ebenfalls einige vorgefertigte Standardelemente, die eine schnelle Implementierung garantieren. Allerdings werden nicht automatisch alle FBML-Elemente auch in XFBML unterstützt. Alle FBML-Elemente (vgl. auch Kapitel 6 und Anhang B) können jedoch mithilfe eines speziellen XFBML-Elements (*ServerFbml*) auch innerhalb von XFBML angezeigt und genutzt werden.

Die folgende Tabelle liefert einen Überblick über die aktuell verfügbaren XFBML-Elemente.

XFBML-Elemente		
Name	Tag	Beschreibung
<i>Activity</i>	<code><fb:activity></code>	Rendert sämtliche Aktivitäten einer Webseite.
<i>Bookmark</i>	<code><fb:bookmark></code>	Zeigt eine Schaltfläche zum Hinzufügen eines Lesezeichens an.
<i>Comments</i>	<code><fb:comments></code>	Fügt ein Kommentarfeld ein.
<i>Connect-Form</i>	<code><fb:connect-form></code>	Erstellt ein Formular zum Einladen von Freunden.
<i>Container</i>	<code><fb:container></code>	Rendert einen Container für einen speziellen Inhalt, der nur sichtbar ist, wenn eine bestimmte Bedingung erfüllt ist.
<i>EventLink</i>	<code><fb:eventlink></code>	Rendert einen Link zu einer Veranstaltung.
<i>Facepile</i>	<code><fb:facepile></code>	Rendert ein Feld mit Profilbildern befreundeter Benutzer.
<i>Like-Box</i>	<code><fb:like-box></code>	Rendert ein Fanfeld.
<i>GroupLink</i>	<code><fb:groupink></code>	Rendert den Link zu einer Gruppe.
<i>Live-Stream</i>	<code><fb:live-stream></code>	Rendert ein Live-Stream-Feld.
<i>Login-Button</i>	<code><fb:login-button></code>	Rendert eine Schaltfläche zum Anmelden.
<i>Name</i>	<code><fb:name></code>	Rendert den Namen eines Benutzers.
<i>Photo</i>	<code><fb:photo></code>	Rendert ein Foto eines Benutzers.
<i>Profile-Pic</i>	<code><fb:profile-pic></code>	Rendert das Profilbild des Benutzers.
<i>Prompt-Permission</i>	<code><fb:prompt-permission></code>	Rendert einen Dialog zum Einholen zusätzlicher Genehmigungen.
<i>Pronoun</i>	<code><fb:pronoun></code>	Zeigt ein passendes Pronomen für einen Benutzer an.

XFBML-Elemente		
<i>Recommendations</i>	<code><fb:recommendations></code>	Rendert ein Feld mit Empfehlungen für eine Webseite.
<i>ServerFBML</i>	<code><fb:serverFbml></code>	Rendert beliebige FBML-Elemente.
<i>Share-Button</i>	<code><fb:share-button></code>	Rendert eine Schaltfläche zum Teilen eines Inhalts.
<i>Unconnected-Friends-Count</i>	<code><fb:unconnected-friends-count></code>	Zeigt die Anzahl aller Freunde an, die nicht mit dieser Applikation via Connect verbunden sind.
<i>User-Status</i>	<code><fb:user-status></code>	Rendert den Status eines Benutzers.

Aktivitäten

Aktivitäten einer Webseite können mit dem Activity Feed Widget angezeigt werden. Dabei zeigt es sämtliche Inhalte an, die von Benutzern bevorzugt (Like) sowie darüber geteilt worden sind. Sofern der Benutzer eingeloggt ist, werden Aktivitäten seiner Kontakte speziell hervorgehoben. Ist er ausgeloggt, werden alle Aktivitäten von allen Personen angezeigt – zudem erhält der aktuelle Benutzer die Option zum Einloggen.

Mit dem Assistenten im Entwicklerbereich können sämtliche möglichen Einstellungen vorgenommen werden.

The image shows a configuration interface for a Facebook social plugin. On the left, there are several settings: Domain (cnn.com), Width (500), Height (300), Header (checked), Color Scheme (light), Font, and Border Color. A 'Get Code' button is located at the bottom of this sidebar. The main area displays a 'Recent Activity' feed with five items, each with a profile picture, name, and description of the activity, along with a timestamp. The items are: Ken commented on Obama's off-shore oil plan (8 seconds ago), Frial shared Israel reverses its iPad ban (9 seconds ago), Ron shared CNN.com Video (11 seconds ago), Tricia commented on iReporters' most memorable rainbows - CNN.com (14 seconds ago), and Mike shared Taliban suspected of sickening female Afghan students (28 seconds ago). At the bottom of the feed, there is a 'Facebook social plugin' logo.

Mit einem Klick auf *Get Code* kann der benötigte Quellcode zur Integration des Widgets ermittelt werden.

```

■ <iframe src="http://www.facebook.com/plugins/activity.php?
■     site=databecker.de&
■     width=500&
■     height=300&
5     header=true&
■     colorscheme=light"
■     scrolling="no"
■     frameborder="0"
■     allowTransparency="true"
10    style="border:none; overflow:hidden;
■     width:500px; height:300px">
■ </iframe>

```

Sofern innerhalb einer Webseite bzw. Applikation bereits die JavaScript-Bibliothek (vgl. Kapitel 3.5) verwendet wird, kann das Widget auch per XFBML eingebunden werden.

Die Funktionalität dieses Widgets basiert dabei auf dem XFBML-Element *Activity*.

XFBML-Element Activity	<fb:activity>
Rendert einen Aktivitäten-Feed für eine spezielle Webseite.	

```

■ <fb:activity site="databecker.de"
■     width="500"
■     height="300"
■     header="true"
5     colorscheme="light" />

```

„Gefällt mir“-Feld

Passend zur „Gefällt mir“-Schaltfläche (Like-Button) existiert auch ein „Gefällt mir“-Feld, das die Benutzer anzeigt, denen eine Facebook-Seite gefällt (früher bekannt als Fanfeld). Zusätzlich kann allerdings auch der Stream der entsprechenden Seite angezeigt werden. Neue Benutzer erhalten zudem über die integrierte „Gefällt mir“-Schaltfläche die Möglichkeit, die Seite für gut zu befinden.

Sämtliche möglichen Einstellungen können mit dem bereits bekannten Assistenten vorgenommen werden. Die Angabe einer ID der entsprechenden Facebook-Seite ist obligatorisch.

4. Facebook Open Graph

The screenshot shows the configuration interface for the Facebook Open Graph widget. On the left, there are several input fields and checkboxes: 'Facebook Page ID (?)' with the value '170886670402', 'Width (?)' with '292', 'Height (?)' (empty), 'Connections (?)' with '10', 'Stream (?)' with a checked box, 'Header (?)' with a checked box, and a 'Get Code' button. The main area displays a preview of the widget for 'DATA BECKER GmbH & Co KG'. The preview includes a post with a 'Like' button, a featured image, and text: 'Lust auf einen Tapetenwechsel? Drei neue Designs für web to date und shop to date verfügbar'. Below the post, it says '165 people like DATA BECKER GmbH & Co KG' and shows ten user avatars with names: Peter, Günther, Rainer, Steffen, Klaus, Peter, Jörg, Marco, Lothar, Tam. At the bottom of the preview is the text 'Facebook social plugin'.

Mit einem Klick auf *Get Code* kann der benötigte Quellcode zur Integration des Widgets ermittelt werden.

```
■ <iframe src="http://www.facebook.com/plugins/likebox.php?
■       id=170886670402&
■       width=292&
■       connections=10&
5       stream=true&
■       header=true"
■       scrolling="no"
■       frameborder="0"
■       allowTransparency="true"
10      style="border:none; overflow:hidden;
■         width:292px; height:px">
■ </iframe>
```

Die Funktionalität dieses Widgets basiert dabei auf dem XFBML-Element *Like-Box*.

XFBML-Element Like-Box	<fb:like-box>
Rendert ein „Gefällt mir“-Feld.	

- `<fb:like-box id="170886670402"`
- `width="292"`
- `connections="10"`
- `stream="true"`
- 5 `header="true" />`

„Gesichter Menge“ (Facepile)

Das „Gesichter-Mengen“-Widget zeigt die Profilbilder aller befreundeten Benutzer an, die sich bereits auf der aktuellen Webseite bzw. für die aktuelle Applikation angemeldet haben. Für die Anmeldung neuer Benutzer ist dieser Vorgang sehr wichtig, da vorhandene befreundete Benutzer zusätzlich für Vertrauen in die Applikation sorgen und dementsprechend schneller die Entscheidung zur Nutzung der Applikation getroffen wird.

Der Assistent bietet die komfortable Konfiguration der Breite des Feldes sowie die Anzahl der anzuzeigenden Zeilen. Die Höhe und die Anzahl der Profilbilder pro Zeile ermittelt das Widget selbst.

Die Funktionalität dieses Widgets basiert auf dem XFBML-Element *Facepile*.

XFBML-Element Facepile	<code><fb:facepile></code>
Rendert ein Feld mit den Profilbildern der aktuell registrierten Freunde eines Benutzers.	

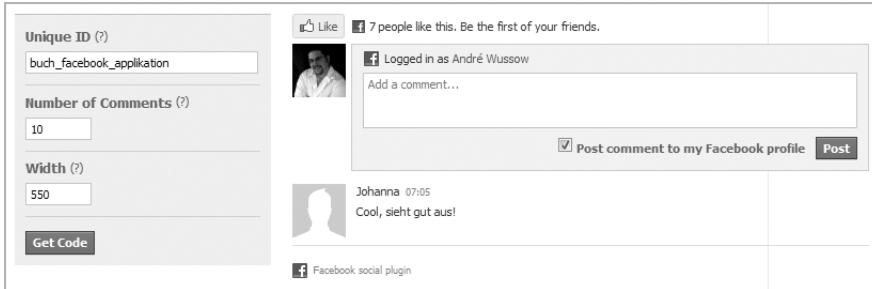
- `<fb:facepile max-rows="1" width="200"></fb:facepile>`

Kommentarfeld

Sehr hilfreich ist auch das Kommentarfeld. Eingebunden ermöglicht es Nutzern, Kommentare oder Rezensionen zu schreiben, aber auch allgemeine Fragen zu einem Inhalt zu stellen. Von Vorteil ist zudem, dass auf eigener Seite nichts zusätzlich (Datenbank, Rechtemanagement, Verwaltung etc.) erstellt werden muss – die vollständige Verwaltung der Daten übernimmt Facebook.

4. Facebook Open Graph

Darüber hinaus können Inhalte im Profil bzw. Stream des Kommentators gespeichert werden, was die Viralität deutlich erhöht. Der Assistent liefert den Code, um das Feld in eine eigene Seite bzw. Applikation zu integrieren.



Die Funktionalität dieses Widgets basiert auf dem XFBML-Element *Comments*.

XFBML-Element Comments	<fb:comments>
Rendert eine Standard-Kommentarbox im Facebook-Design.	

Die einfachste Methode, ein solches Kommentarfeld einzubinden, ist das *fb:comments*-Element – ohne Anwendung eines Attributs. Generell empfiehlt es sich, den vom Assistenten generierten Quellcode zu nutzen.

- `<fb:comments numposts="10" width="550"></fb:comments>`

Sofern der Eigentümer der Kommentarbox bzw. ein Administrator der entsprechenden Applikation, in deren Kontext das Kommentarfeld läuft, sich auf der Seite mit dem Kommentarfeld befindet, bekommt er automatisch die erforderlichen Rechte zur Verwaltung der Kommentare und kann gegebenenfalls Einträge löschen.

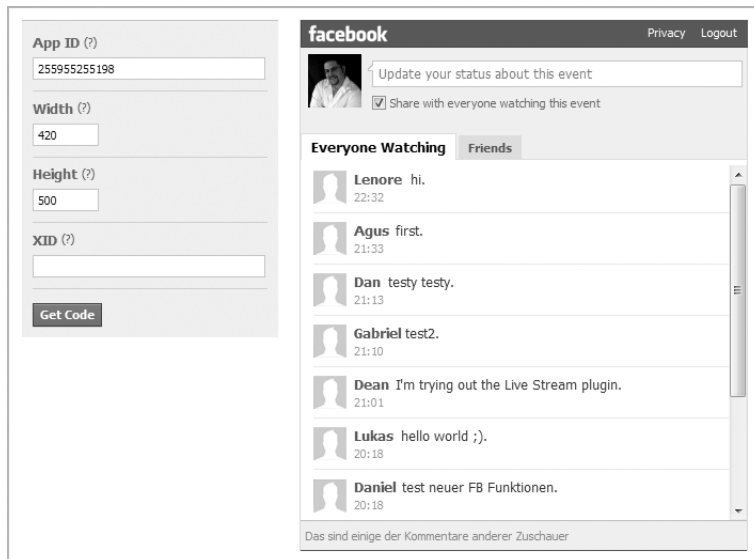


Live-Stream-Feld

Ein weiteres Widget ist das Live-Stream-Feld. Es ermöglicht Besuchern der Webseite bzw. der Applikation, ihre Aktivitäten und Kommentare in Echtzeit zu teilen. Besonders herausstechend ist die Möglichkeit, mehrere Felder für verschiedene Themen oder Veranstaltungen zu nutzen.

Die beste Umgebung für das Live-Stream-Feld stellt eine entsprechende Liveveranstaltung wie beispielsweise ein Livekonzert, eine Konferenz, ein Webcast oder ein Multi-Player-Spiel dar.

Sämtliche möglichen Einstellungen können mit dem bereits bekannten Assistenten im Entwicklerbereich vorgenommen werden. Die Angabe einer ID eines Objekts ist Pflicht. Sofern mehrere Felder auf einer Seite genutzt werden sollen, kann eine eindeutige ID (XID) pro Feld definiert werden.



Mit einem Klick auf *Get Code* kann der benötigte Quellcode zur Integration des Widgets ermittelt werden.

- `<iframe src="http://www.facebook.com/plugins/livefeed.php?`
- `app_id=255955255198&`
- `width=420&`
- `height=500"`
- 5 `scrolling="no"`
- `frameborder="0"`

4. Facebook Open Graph

- allowTransparency="true"
- style="border:none; overflow:hidden;
- width:420px; height:500px">

10 </iframe>

Das Widget kann auch per XFBML eingebunden werden. Die Funktionalität basiert dabei auf dem XFBML-Element *Live-Stream*.

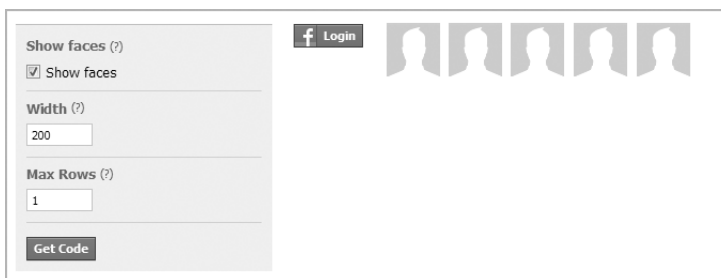
XFBML-Element Live-Stream	<fb:live-stream>
Rendert ein Live-Stream-Feld.	

- <fb:live-stream app_id="255955255198"
- width="420"
- height="500" />

Login-Schaltfläche

Das Login-Widget verbindet die Möglichkeit zur Nutzung der Applikation mit den Vorteilen des „Gesichter-Mengen“-Widgets. Neben der einfachen Login-Schaltfläche zeigt es die Profilbilder aller befreundeten Benutzer an, die sich bereits auf der aktuellen Webseite bzw. für die aktuelle Applikation angemeldet haben.

Der Assistent bietet auch hier wieder die komfortable Konfiguration der Breite des Feldes sowie die maximale Anzahl der anzuzeigenden Zeilen. Die Höhe sowie die Anzahl der Profilbilder pro Zeile ermittelt das Widget selbst.



Die Funktionalität dieses Widgets basiert auf dem XFBML-Element *Login-Button*.

XFBML-Element Login-Button	<fb:login-button>
Rendert eine Login-Schaltfläche mit den Profilbildern bereits angemeldeter befreundeter Benutzer.	

- <fb:login-button show-faces="true" width="200" max-rows="1">
- </fb:login-button>

Empfehlungen

Ein Widget, um einen Benutzer auf Inhalte einer beliebigen Webseite aufmerksam zu machen, ist das Empfehlungs-Widget. Dabei werden sämtliche Inhalte geladen, die zu der angegebenen URL passen und von verschiedenen Benutzern, vorzugsweise von Freunden, auf Facebook veröffentlicht worden sind.

Sämtliche möglichen Einstellungen können mit dem Assistenten im Entwicklerbereich vorgenommen werden. Die Angabe einer Domain, für die Empfehlungen ausgesprochen werden sollen, ist Pflicht.

Domain (?)

Recommendations

Facebook social plugin

Width (?)

Height (?)

Header (?)

 Show header

Color Scheme (?)

Font (?)

Border Color (?)

Mit einem Klick auf *Get Code* kann der benötigte Quellcode zur Integration des Widgets ermittelt werden.