

Web-based mobile Augmented Reality: Developing with Layar (3D)

Heike Sperber
MSc student
hasperb@yahoo.com

Peter Curet
MSc student
petercuret@gmail.com

Klaas-Jan Govaart
MSc student
kgovaart@liacs.nl

ABSTRACT

The Layar Reality Browser is an application at the forefront of the young and promising field of Augmented Reality for mobile devices. This document provides a concise overview of the history of the field as well as an overview of the Layar platform architecture. We reflect on some of the connotations of the development of mobile Augmented Reality, and on its current technological limitations. The getting started section presents a step by step introduction to development for the platform.

1. PURPOSE, CONTEXT AND HISTORY

Increasingly sophisticated mobile devices have opened up the possibility of handling computationally expensive applications on the go. The availability of video capturing, GPS tracking and Internet access in smart-phones has led to the migration of Augmented Reality (AR) systems to these mobile platforms. A widely acknowledged definition of AR was given by Ronald Azuma in 1997 [1]. He defined AR as combining the real and the virtual, interactively and in real time, where the virtual information is registered to the physical world in 3D.

The Layar Reality Browser [2] runs on Android OS and iPhone smart-phones. It captures a stream of images from the phone's built-in camera and displays an additional layer of information on top of it. Users can choose from a range of many possible layers to be overlaid on their environment, enhancing it with additional information. This allows users to quickly find whatever it is they are looking for, whether that be an ATM machine, a restaurant with corresponding reviews, or property for sale.

Using Navstar-GPS [3] coordinates, combined with data from the phone's built-in compass, the Layar browser dynamically retrieves information specific to the current location and orientation of the user through an Internet connection. Information in a layer is defined by 3rd party developers as Points Of Interest. A Point Of Interest (POI) consist of at least a longitude/latitude coordinates pair, combined with additional information such as name and description. Custom icons, links to images and other types of information can be specified as well.

Since Layar offers a open, though moderated, platform for developers to create and add layers to the browser, the number of specific layers that user can choose from is

growing rapidly.

The Layar system was only launched in 2009, but some of the key technologies that underlay it go back some two decades. Here is a list of milestones in the development of mobile AR that have led up to the development of the Layar system.

- 1992: 1st smart-phone, boasting 1MB of memory and a B/W, 160x293 pixel touch screen. It works as a phone, pager, calculator, address book, fax machine, and e-mail device. It weighs 500 grams and costs \$900. [4]
- 1993: The Navstar-GPS [3] system achieves initial operational capability.
- 1997: Phillipe Kahn [5] invents the camera phone.
- 1999: Benefon produces the 1st phone with GPS. [6]
- 2000: The 1st commercial camera phone, by Sharp. [7]
- 2001: Kooper and MacIntyre create the RWWW Browser. It is the first AR browser. [8]
- 2003: The Siemens SX1 [9] phone comes with the first commercial AR camera game. It is called Mozzies.
- 2008: Mobilizy launches Wikitude [10], an application that combines GPS and compass data with Wikipedia entries.

2. OPERATING PRINCIPLES

In the following section the key concepts of the Layar technology will be described. An overview of the system architecture is given, along with a description of what happens during a typical browsing session using Layar.

Another element relevant to developers is the API provided by Layar, which will be highlighted in the last section.

2.1. Architecture Overview

The Layar architecture consists of five logical entities, as illustrated in figure 1.

1: The Layar App is the client, the browser running on the user's smart-phone.

2: The Layer Server is at the heart of the service, mediating clients and external servers, and serving POIs to the clients.

3: The Layar Provisioning Website is where developers submit new layers and manage existing layers and accounts.

4: The Layar Service Providers are servers maintained by developers. These serve POIs to the Layar Server, but can serve POI information for a layer directly to clients as well. This is where a database for a given layer would be.

5: Content Sources serve additional POI information, such as images, directly to the client. They might be integrated with the Layar Services Providers or may be separate entities, in case of large volumes of data.

These independent entities are interrelated:

- The Layar App displays a gallery of layers obtained from the Layar Server.
- Developers can submit their layer to the Layar Server through the Layar Provisioning Website, pointing it to the location of the Layar Service Provider.
- The Layar Service Provider accepts POI requests from the Layar Server, as well as providing additional POI information directly to the client.

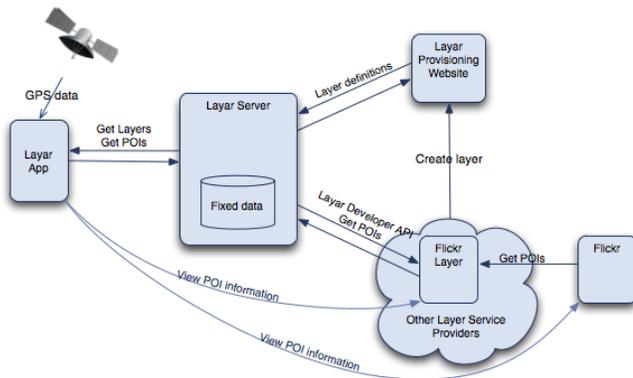


Figure 1. The interrelated entities and processes of Layar.

2.2. Session Description

In a typical session, the client sends and receives information according to its location. As the client accesses a layer, it communicates directly with the Layar Server. The current location of the client is then sent to the server. Subsequently, the server returns POIs in the range of the position of the client, which displays these as an overlay.

The positions of both the client and POIs are determined and stored using the longitude and latitude values of the GPS coordinate system, disregarding the altitude value. Once received, coordinates of POIs are mapped relative to the client to be displayed correctly on screen.

The POIs are content specific and depend on what layer the user has chosen in the Layar browser. The server itself does not store each POI, but acts as a gateway between the client and servers that host that specific layer. In essence, the Layar Server is a central provisioning system, providing clients with all the developed and approved layers. The Layar Server communicates with the servers hosting these

layers, receiving POIs and their information and sending these to the Layar browser on the client.

In short, a session adheres to the following sequence:

- The user selects a layer of interest in the Layar App.
- The Layar App sends the current GPS coordinates to the Layar Server and request POIs for the selected layer.
- The Layar Server requests the latest POIs from the Layar Service Provider for that layer.
- The Layar Server sends POIs to the Layar App.
- The Layar App requests additional POI information directly from the Layar Service Provider or Content Server.
- The Layar App displays the received data as an overlay on the camera input.

2.3. The Layar Developer API

In order for the Layar server to communicate with external sources, two APIs have been defined, namely the Layar client API and the Layar developer API.

The client API provides an interface for the client to communicate with the server, and is not publicly accessible.

The developer API, however, is accessible, and provides an interface for communication between the Layar Server (LS) and the Layar Service Providers (LSP). Using the API, the LS can retrieve current information about the layer.

Information is requested by the client using a HTTP GET call `GetPointsOfInterest[xXx]`. In doing so, the necessary parameters such as client location and authorization information are sent to the LSP, via the LS.

The LSP handles the get call, processing the request parameters and gathering the relevant POIs and their corresponding information from a database. The database is hosted on the same server as the LSP and contains all POIs and necessary data that defines the layer. The LSP then converts the collected POIs into JSON (JavaScript Object Notation) according to the structure defined in the API. This notation is machine interpretable and provides a standard in which to represent and process information, in this case the POI. The request is then completed by sending the JSON data back to the central server and subsequently to the client. PHP is used on the LSP to handle the requests, along with a MySQL database for storing and retrieving POIs.

POIs can include detailed information about locations and the manner in which this information is displayed on the client. Until recently, POIs were indicated with 2D icons. Layar now supports the display of 3D objects as an overlay on the screen. These settings are all defined by the LSP through the JSON data. Additional detailed information

such as texts and pictures can be provided by external sources such as Flickr or Twitter, since these sources also provide geo-coded API services and can be accessed from LSP using the location coordinates of either the client or a POI. Any additional information that might be available from these external sources can be coupled to the POI and returned in the JSON description. The client will then be able to display the external information by accessing these sources directly.

In summary, the developer API exposes the layer data that is provided by the Layar Service Provider, allowing the Layar Server and the Layar Service Provider to exchange information.

3. STRENGTHS AND WEAKNESSES

There is a lot to be said in favor of the Layar platform. Although skeptics doubt whether Layar adds more than just a more entertaining experience of the same information [11], the AR view is in fact more intuitive than a traditional map view for many people. Besides, in cases where a map view of information is preferred, Layar offers that view as well.

The core strength of Layar has nothing to do with AR view vs. map view vs. list view of information, though. On the Layar platform, data is continuously pushed to the user dynamically, depending on location. That is a very powerful and practical approach to filtering data. After selecting a layer of interest at the beginning of the session, users only see the information relevant to them, with the system constantly adapting to location (GPS coordinates) and orientation (compass). Layar offers easy integration with 3rd-party content providers (e.g. Flickr, Twitter) which makes this system even more powerful.

Another great advantage of Layar is the fact that it provides an open platform for any developer, whether commercially or artistically motivated, at no cost. Developers can even make money using Layar by commercially offering their creations. Through three dimensional objects, sounds linked to POIs, easter-eggs based on proximity, and even POIs distinguishable by altitude tags, in case they overlap, the Layar platform offers many possibilities for a rich user experience.

A drawback to Layar are the high initial costs. Layar requires an expensive smart phone with camera, GPS and compass to run in addition to a high bandwidth. This at the moment somewhat limits the target audience though efforts are being made to expand to other platforms [12].

Since Layar is still a relatively young product it continues to be a work in progress with issues that need to be resolved. This holds especially true for any new feature that is introduced e.g. the ability to display 3D objects, which currently places severe restraints on the 3D model and its complexity [13]. In general Layar still suffers from

performance problems which lead to limitations in real-time rendering [14]. Besides the phone's processor the technology also depends on the strength and accuracy of the GPS signal. Currently it can easily vary between 20-50m, which is not exact enough when pointing the phone at a single building. Some applications compensate by building into their layer the possibility to adjust one's position to improve the result [15].

Though Layar offers an open development platform to create layers, the technology itself is not open source. To make a layer publicly viewable it needs to be submitted and reviewed by Layar. The criteria presently only address technical as well as usability issues [16] however the process will be expanded to meet future requirements, which leaves the possibility for rejection based on content. Another negative aspect are Layar's close ties to Google, a company that has repeatedly been under fire for amassing personal user data without disclosing what it will be used for [17,18].

4. INTENDED APPLICATIONS

Approximately 650 layers live and 2500 in development exist for Layar today (May 2010) [19]. At the forefront are the basic information-based applications and popular web services. The former let you view real-estate listings as you walk along the street [HomeScan, 20], read reviews and see the price range for a restaurant you are standing in front of [Scope, 15], find the closest ATM [21] or educate yourself about the sights in a city you are visiting [22, 23 (figure 2)].



Figure 2. The Eytour layer.

However the range also extends to more specialized layers such as Spotcrime in the US that displays current crime statistics based on your location [24] or a Walt Disney World guide to all the attractions and related services in the theme parks [25].

Among the well-known internet sites that extend their reach to the mobile phone are Youtube, Twitter, Flickr, Foursquare and Wikipedia that allow you to view geotagged tweets [26], videos or photos in the physical

space around you or simply include the user's location in their usual services.

Whereas some layers simply present information, others offer more interactivity. Recently advertising agencies have discovered Layar as a new medium. Especially the entertainment industry is starting to offer games and gadgets relating to their latest production. The launch of the video game Splintercell was accompanied by a virtual hunt around the city where players took on the characters' personas and went on various missions to find clues only visible on their smart phones [27]. Walt Disney tied a Layar campaign to the movie posters for their new movie Prince of Persia including a quiz and theatre locator [28]. A layer produced for the movie Avatar places faces of the characters in one's surrounding, which can be overlaid on real people like a mask and saved as a picture screenshot [29, figure 3]. A similar concept adds celebrities or fictional characters into a photo taken of a person [29].

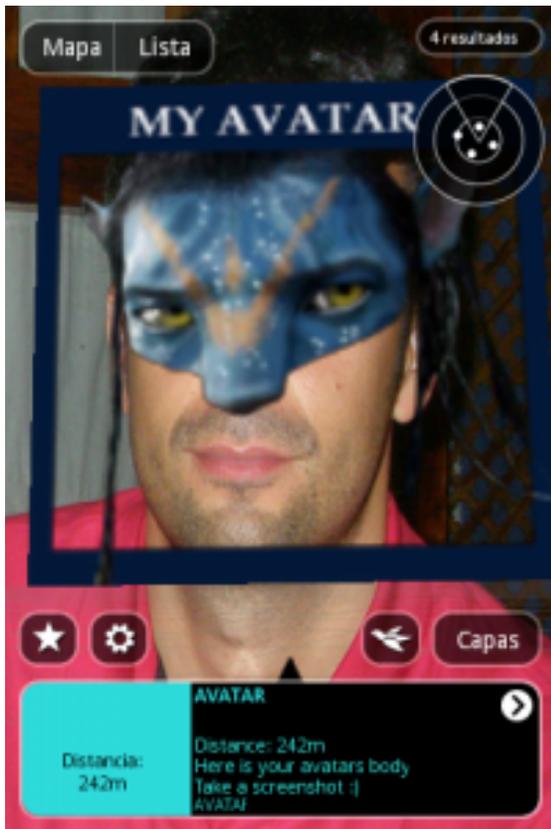


Figure 3. Avatar; using Layar for entertainment.

A further development lets users place content: they can add a new restaurant or spot in the city [30] or leave 3D easter eggs with messages for their friends to find [31].

Another sector that is making its way into mobile AR are the arts. At the first virtual 3D-exposition in the Netherlands "Ik op het Museumplein" [32] by MediaLAB/Stedelijk Museum in Amsterdam (May 28 - June 30, 2010) the visitors can see student art works only

with the help of their smart phones. The intent is to explore how the new medium changes the nature of a piece of art. With spatial and physical restraints lifted the questions the initiators addressed to the young artists were "What if you can put your artwork anywhere on the streets? What would your artwork look like?"

On the other hand the Virtual Public Art Project (VPAP) placed 14 objects around the globe that are explicitly meant to be site specific despite its virtual portability [33].

Besides creating new content the technology also lends itself to reviving lost objects or past events. The WTC Layer resurrects the World Trade Center when the phone is pointed on its former location [33]. The Berlin Wall can now be revisited around the Brandenburg Gate including several watch towers and the corresponding historical information [34].

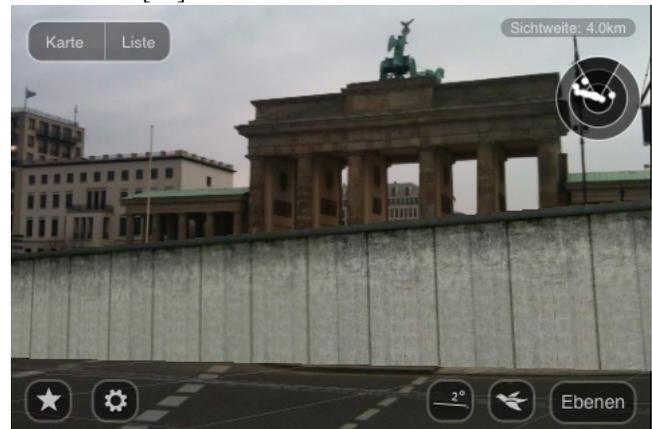


Figure 4. Time traveling? The Berlin Wall resurrected.

This selection represents just a fraction of an ever-increasing catalog of layers that with increasing popularity and technical improvement will conquer new fields and applications.

5. UNINTENDED APPLICATIONS

Given that Layar is only about to celebrate its first birthday there has not been much time for the technology to be adapted for other not initially intended uses. One might speculate though whether the founders of Layar envisioned such a wide-spread application beyond the obvious information-driven services.

As with every new medium the question arises how to deal with the political and ethical implications. Is access to the technology to be restricted or will there be a Nazi sightseeing tour of Amsterdam [35]? At the moment all new layers have to be approved by the Layar company, yet if the market will expand as expected, access and control can not realistically remain fully centralized. This means some form of government regulation would probably soon be a necessity. The debates about restrictions and censorship of digital media will have to include mobile AR.

6. GETTING STARTED

This section will briefly describe the development of a basic layer. The details and the example code can be found in the reference tutorials by Layar [36] and Stedelijk Museum [37].

6.1. Requirements

In terms of hardware a web server with PHP (5.2 or above) which supports JSON is required as well as a MySQL database (preferably with phpMyAdmin). The following guide uses PHP but layers may be programmed in a variety of programming languages. Layar itself uses JSON.

Before getting started you need to request a developer key once which requires a Google account. Register at <http://dev.layar.com> and you will receive an e-mail with an activation link within a few days. Once you log in with your Google password the two accounts will be linked.

In order to properly test your layer before submission you will need an iPhone 3GS or Android based phone (with GPS and compass) with the developer APK installed [38].

6.2. Step-by-step instructions

6.2.1. Define a layer on the publishing site

After signing in to your new account click on ‘Create new layer’ and fill out the form as seen in Figure 5 (Add a layer). As type choose ‘Generic 2D’. The POI URL is the location of the php file we will create later that retrieves the data for the POIs.

Figure 5. Add a layer.

Click on ‘Save and customize’ to add more details below the tabs. In the general section for example you can restrict where users will be able to see your layer and upload an icon for your layer under ‘Listing’. ‘AR View’ allows you to adjust the layout and colors of the interface of the layer.

6.2.2. Prepare the database

The next step is to create a database that holds the information about our POIs. Both tutorials provide the SQL code to create insert an empty table for the POIs.

6.2.3. Gather POIs information

The database table now needs to be filled with the coordinates of the POIs. On Google Maps find the location you would like to include and right-click on it. Select “What’s here” and Google will display the GPS coordinates of the position in the search field. Add the coordinates in the lat and long fields in a new entry into the database and complete the additional information (Figure 6).

| Field | Type | Function | Null | Value |
|-------------|----------------|----------|-------------------------------------|---------------------------------------|
| id | int(50) | | | 1 |
| attribution | varchar(50) | | <input type="checkbox"/> | The location of Layar Office |
| title | varchar(50) | | <input type="checkbox"/> | Layar office location |
| lat | decimal(20,10) | | <input type="checkbox"/> | 52.3741180000 |
| lon | decimal(20,10) | | <input type="checkbox"/> | 4.9342500000 |
| imageUrl | varchar(255) | | <input type="checkbox"/> | http://custom.layar.nl/layarimage.jpg |
| line4 | varchar(50) | | <input type="checkbox"/> | 1019DW Amsterdam |
| line3 | varchar(50) | | <input type="checkbox"/> | distance:%distance% |
| line2 | varchar(50) | | <input type="checkbox"/> | Rietlandpark 301 |
| type | int(11) | | <input type="checkbox"/> | 1 |
| actions | varchar(50) | | <input checked="" type="checkbox"/> | |
| dimension | int(1) | | <input type="checkbox"/> | 1 |
| alt | int(10) | | <input checked="" type="checkbox"/> | |
| relativeAlt | int(10) | | <input checked="" type="checkbox"/> | |
| transform | int(10) | | <input checked="" type="checkbox"/> | |
| object | int(10) | | <input checked="" type="checkbox"/> | |
| distance | decimal(20,10) | | <input checked="" type="checkbox"/> | |

Figure 6. POI table.

To allow the layer to load and display information from the database we need a PHP script to establish a link. It needs to handle a request from the layer and determine which information to query the database for. The retrieved data has then to be translated into JSON, the operating language of Layar and sent back to the layer where the user can view the result. Again, the tutorials provide example code to try out or modify.

6.2.4. Test the layer

The layer can be tested from within the API by placing a test person on the map and clicking ‘Load POIs’. A better field test though obviously is to use the layer as your users will experience it later. With the developer APK installed on a supported phone you can add your developer key in the application settings and subsequently find your layers in development listed under the ‘Featured’ tab. It is also possible to choose a different location for the phone to test POIs that are not within the vicinity.

6.2.5. Publish the layer

A layer needs to be submitted for approval before it will be published and becomes visible to the general public. In the API under 'Your layers' click the 'Request publication' button next to the layer. It will take a few days for Layar to review your layer. The rejection rate is around 50% for technical reasons [39] therefore it is important to test your layer thoroughly before submitting it to avoid delaying the process.

6.3. Further information

The tutorial by the Stedelijk Museum [37] already includes some advanced features such as auto trigger actions and objects. Layar's platform wiki [40] is a good source of information on the inner workings of the technology and its features including links to tutorials. A developer group [41] has been established to facilitate exchange and discussion.

7. FINAL THOUGHTS

Mobile AR is a versatile technology that offers real-time information, new forms of interaction and creative applications. Since the relative ease of development allows a broad base of users to create their own layers the acceptance and use of Layar will only continue to spread. Third party tools [42] that will further simplify the process can even more increase the diversity of the catalog. On the negative side this then leads to the cluttering of the phone screen with POIs. Layar just introduced Layar Stream which serves to filter layers by user preferences as well as general popularity and usage [43]. Further problems such as GPS inaccuracy will have to be dealt with as well. The long anticipated European Navstar equivalent Galileo [44] and its commercial service with accuracy of up to 1 centimeter (0.4 inch) might play a role in future developments. But even so, experts predict that it will not be long before AR on the mobile phone is superseded by AR integrated into wearables [45].

REFERENCES

1. Azuma, R., "A Survey of Augmented Reality", Presence: Teleoperators and Virtual Environments, 1997, pp 355-385.
2. Layar Reality Browser, <http://layar.com>
3. Navstar-GPS, http://en.wikipedia.org/wiki/Global_Positioning_System
4. Simon smart-phone, [http://en.wikipedia.org/wiki/Simon_\(phone\)](http://en.wikipedia.org/wiki/Simon_(phone))
5. Phillippe Kahn, http://en.wikipedia.org/wiki/Philippe_Kahn
6. Benefon Esc! NT2002, <http://www.benefon.de/products/esc/>
7. Sharp J-SH04, http://k-tai.impress.co.jp/cda/article/showcase_top/3913.html
8. Kooper, R., Browsing the Real-World Wide Web: Maintaining Awareness of Virtual Information in an AR

Information Space, International Journal of Human-Computer Interaction, Vol. 16, Nr. 3, 2003, pp. 425-446.

9. Siemens SX1, http://en.wikipedia.org/wiki/Siemens_SX1
10. Mobilizy, <http://en.wikipedia.org/wiki/Wikitude>
11. <http://www.nytimes.com/2010/03/26/greathomesanddestinations/26iht-rear.html?scp=3&sq=layar&st=cse>
12. <http://site.layar.com/download/supported-devices/>
13. <http://layar.pbworks.com/Creating-the-3D-objects>
14. <http://stedelijk.medialab.hva.nl/2010/03/09/impression-of-visible-cities-at-trouw-amsterdam/>
15. <http://bits.blogs.nytimes.com/2009/10/14/urbanspoon-makes-it-easier-to-scope-out-restaurants/>
16. <http://layar.pbworks.com/The-publication-process>
17. <http://news.bbc.co.uk/2/hi/technology/10172146.stm>
18. <http://edition.cnn.com/2010/TECH/social.media/06/01/cnet.congress.facebook.privacy/index.html?iref=allsearch>
19. <http://site.layar.com/company/blog/finding-layar-media/>
20. <http://www.nytimes.com/2010/03/26/greathomesanddestinations/26iht-rear.html?scp=3&sq=layar&st=cse>
21. http://www.businessweek.com/magazine/content/09_47/b4156052745487.htm
22. <http://site.layar.com/company/blog/layar-launches-worlds-first-augmented-reality-content-store/>
23. <http://eyetour.com/blog/tag/layar/>
24. <http://spotcrime.blogspot.com/2010/05/spotcrime-is-now-on-android.html>
25. <http://mousereality.com/wdw.html>
26. <http://allyourlayarsarebelongtous.com/tweeps-around.html>
27. <http://www.youtube.com/watch?v=8URL44EM5Os>
28. <http://site.layar.com/company/blog/layer-of-the-week-prince-of-persia/>
29. <http://site.layar.com/company/blog/20-new-layers-launched-have-fun-with-screenshot-layers/>
30. <http://site.layar.com/company/blog/300th-layer-launched-find-tourist-locations-in-australia-the-united-states-and-ten-european-countries/>
31. <http://www.hoppala.eu/layer/hoppala-goes-easter>
32. <http://www.ikophetmuseumplein.nl/>
33. <http://site.layar.com/company/blog/32-new-layers-launched-get-inspired-by-art-culture-layers/>
34. <http://www.hoppala.eu/>
35. Bruce Sterling, "At the Dawn of the Augmented Reality Industry". [@ ±30 minutes.](http://vimeo.com/6189763)

36.<http://layar.pbworks.com/First+Layar+Tutorial+-+Create+a+simple+layer>

37.<http://stedelijk.medialab.hva.nl/2010/03/02/creating-a-layar-layer-a-step-by-step-tutorial/comment-page-1/#comment-119>

38.<http://layar.pbworks.com/Download-the-APK>

39.<http://site.layar.com/company/blog/layar-developers-and-publishers-please-test-your-layers-before-you-request-publication/>

40.<http://layar.pbworks.com>

41.<http://groups.google.com/group/layar-developers?hl=en>

42.<http://site.layar.com/create/tools/>

43.<http://site.layar.com/company/blog/layar-announces-layar-stream/>

44.<http://www.esa.int/esaNA/galileo.html>

45.<http://www.nytimes.com/2009/07/12/business/12proto.html>