

Connecting the dots between Angular single page apps and Magic xpa

Content

What is a Single Page Application?	» 2
What skillsets does your organization need in order to utilize Magic's web application framework?	» 3
What are the Business advantages of Magic's web application framework?	» 4
What are the technical advantages of Magic's web application framework?	» 5
What are UI services and how do they fit into the framework?	» 6
Why move to Magic's web application framework - A summary.	» 7

What is a Single Page Application?

A single-page application (SPA) is a web application or website that interacts with the user by dynamically rewriting the current web page with new data from the web server, instead of the default method of the browser loading entire new pages. The goal is faster transitions that make the website feel more like a native app.

In a SPA, all necessary HTML, JavaScript, and CSS code is either retrieved by the browser with a single page load, or the appropriate resources are dynamically loaded and added to the page as necessary, usually in response to user actions. The page does not reload at any point in the process, nor does it transfer control to another page, although the location hash or the HTML5 History API can be used to provide the perception and navigability of separate logical pages in the application.

(Wikipedia)

What Is Angular?

Angular is an open-source JavaScript framework developed and maintained by Google that helps developers and designers build dynamic web applications using simple HTML tags. The applications can then be run on both desktop and mobile devices.



Do Magic and Angular work together?

It's as if Magic had been waiting all of these years for a technology like Angular to come around so that Magic can best show off its development paradigm.

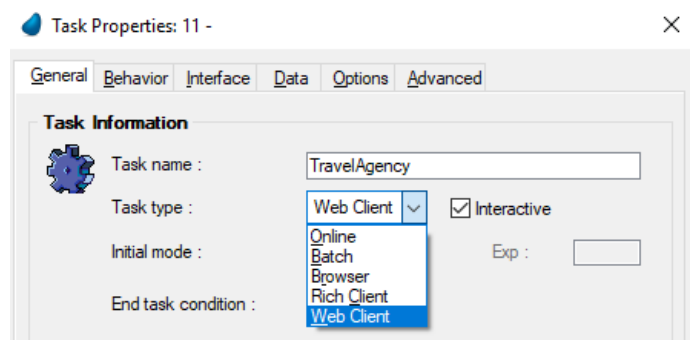
By working with Magic to develop your Angular applications, you benefit from the best of both worlds since the two have a full synergetic fit. You define the business logic in Magic, both for the backend and for the frontend, and develop the presentation using the Angular code. This synergy simplifies coding in Angular using Magic's technology and philosophy. It's all about making life easier for the developer.

Applications created with Magic's web application framework will be pure Angular applications, meaning they will be deployed and managed exactly like any Angular application.

What skillsets does your organization need in order to make use of Magic's web application framework?

Magic xpa

It's a quick learning curve, since the client is designed very much like the usual Magic xpa Online and Rich Client programs. You develop the same way you are familiar with and then use a new task type within Magic called Web Client.



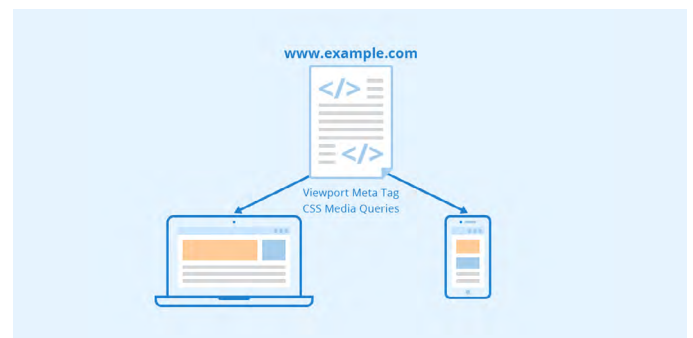
Angular and/or JavaScript

As a Magic developer, you only need to know basic Angular. Once the Angular code is generated from Magic, the Angular part is over. You only need to understand the connectivity between Magic and Angular and how Magic's engine uses Angular's template syntax. Angular's goal is to let you extend the vocabulary of HTML, which you can do by defining your own tags. This ability to extend the vocabulary is how Magic syncs with Angular and helps you build the extensible HTML tags much more productively than the raw Angular implementation.

```
<input
  *ngIf="mg.isRowInRowEditing(row)"
  matInput
  [magic]="mgc.MessageID"
  [rowId]="row.rowId"
  [formControlName]="mgc.MessageID"
  mgFormat
```

Web Design

Once the Magic directives have been added to the HTML code, you will have functioning screens. Any external IDE can then be used to design the screens. Anyone who knows HTML and CSS will be able to design the screens. About 90% of all the modifications needed will be to the HTML and CSS. Only about 10% of the work will be in Angular and/or JS.



What are the advantages of using Angular as opposed to a Rich Client module, at the frontend?

Your apps are client-free since the deployed application is pure JS, HTML5, and CSS3. The apps can be accessed directly without having to pre-install clients. The freedom of HTML design is unlimited, and you can outsource the design of the screens to web designers and web developers in your organization.

What are the business advantages of Magic's web application framework?

Angular is an open-source JavaScript framework developed and maintained by Google that helps developers and designers build dynamic web applications using simple HTML tags. The applications can then be run on both desktop and mobile devices.

✔ **Application Development and Delivery**

Magic's everlasting advantages are relevant now, more than ever. Magic xpa's development paradigm offers a remarkably fast time to market while utilizing a small but high-quality team of developers. Maintenance of large-scale applications becomes easy and low priced.

✔ **Advanced Customization of Your UIs**

The beauty of decoupling the UI and the logic, which Magic's web application framework makes possible, is the flexibility to split the effort and time invested, between the business logic and the UI/UX. The two parts are equally important and time consuming, but when you have such an effective way of splitting the two you can achieve better results. The work can be done by the same developer (assuming the developer has the required skills) or by two different developers. For the UI/UX part, the developer does not require any Magic programming knowledge.

✔ **Ease of Maintenance**

You can change your application logic from Magic and it will be reflected automatically in your Angular-based apps. Thus, there is less effort to make changes to your application in future.

✔ **IP Value**

Throughout the years Magic xpa has proven itself as a future proof platform, allowing the application owner to continue using their well-known and reliable IP, when moving between Magic versions and between deployment architectures. This is extremely relevant when considering a switch to the web application framework.

What are the technical advantages of Magic's web application framework?

✔ State-of-the-Art Technology

Magic's Web application framework uses state-of-the-art client-side technologies including JavaScript, Angular, CSS3, HTML5, and TypeScript. You can create full-fledged business web-based applications that are heavy-duty, including data entry, transaction intensive, enabling you to integrate your applications with everything out there.

✔ Magic xpa's Client-Side Architecture

All of the UI-related work, including the UI logic, is done by the browser (the client), which enables you to create fast and dynamic web apps. In this architecture, the server only sends raw data and application logic to the client and the client generates the screen. The browser encompasses three layers:

- 1 Angular coding, including HTML, CSS and TypeScript files. This is different for each application.
- 2 Angular API. This is generic and the same for each application.
- 3 Engine. This is also generic.



✔ Less Code; More Versatility

The Magic engine's robust business logic paradigm can be leveraged to save a lot of code in Angular by setting up most of the behind-the-scenes logic for the dynamic display of the presentation of the UI. In practice, Magic xpa offers a wide range of control and form properties, which are very simple to use and can save a lot of lines of coding in Angular - for example, validation, recompute, conditions, visible, enabled, tooltips, and more.

Expressions on any of the properties are very easy to use in Magic, which otherwise would take dozens of lines of codes to develop in Angular, especially when you are trying to achieve dynamic behavior in your apps.

Can Magic's web client applications be deployed in the cloud?

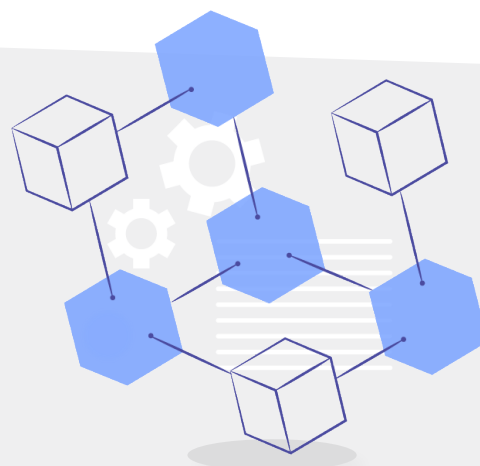
The web client runs in a browser, so that anyone with internet access can run an application deployed by this framework. Keep in mind that the application running within the browser is a 100% **standard Angular application**, with a Magic backend. Therefore, all you need to do is to deploy the backend modules in the cloud, and your Angular application will be cloud-based.

Which browsers are supported when using the web application framework?

Your web applications will be platform agnostic. It will run on any modern browser and on any device with zero footprint. You do not need to install anything.

Can a Client/Server or a Rich Client application be migrated into this framework?

Both client/server and rich client apps can easily be migrated to web client. The web client development methodology is similar to the rich client methodology, therefore allowing an even quicker migration path.



What are UI services and how do they fit into the framework?

UI services is a term that Magic introduced. The rationale is that when you examine a web application you can split it into a frontend and a backend, but this would be a technical split. We split the application into **application logic** and **presentation logic**.

All application/business logic should be handled by the application developer (regardless of where it is running – on the server or on the client) and presentation logic should be handled by the UI/UX developer mainly implementing CSS.

Magic UI Services are all the services that the application developer needs to provide to the UI developer to implement the UI/UX.

For example: invalid value in a control, the validity or invalidity will be determined by the application logic developer but how the control will look when it is invalid will be implemented by the UI/UX developer.

Can a Web Client application be deployed on mobile clients?

Of course, since Magic Web Client is a pure Angular application can be deployed on any device, as Angular can be deployed on any device.

Why move to Magic's web application framework? – A Summary

Category	Benefits
New UI/UX	<ul style="list-style-type: none">» Web designer friendly: You can easily create new, intuitive and modern screens for your apps and you don't have to design the screens yourselves. The previous versions of Magic required you to design the screens. You can now let Web designers handle this part of the application - freeing you up to do what you do best.
State of the art- SPA	<ul style="list-style-type: none">» SPA-compliance: Magic's web application framework is way ahead of the market. This way of programming will be commonplace in the future, in the next 2-5 years.
Business logic portability	<ul style="list-style-type: none">» You can migrate your Magic business logic, while keeping the logic intact, from previous Magic task types, such as Online and Rich Client, to the new Web Client task type.» You can use the business logic regardless if it is on the server or client.
Browser-based deployment	<ul style="list-style-type: none">» Zero footprint: You and the customer don't need to install anything.» The app will run on all devices.» Pure Angular application. Deploy and manage it like any other Angular application
Skillset	<ul style="list-style-type: none">» No additional Magic knowledge is needed. You already know what you need to know in Magic.
Cloud enabling	<ul style="list-style-type: none">» Easy access for everyone from the cloud to a demo or a free trial.» Less deployment time. Just send a link and a password to the customers without having to set-up any environment.» Although an initial investment is required, it will be more cost-effective in the long run.

Contact us for more information