Wildcard URLs provide a way to load content dynamically depending on the page URL.

You can find an example on the sample Community starter site. The **Members -> Profile** page uses wildcard URLs to display user profiles — a single page displays profiles of various site users.

How is it achieved? Log in to the administration interface and open the **Pages** application. Select the **Members -> Profile** page in the content tree and switch to the **Properties -> URLs** tab. You can see **/Members/{UserName}** in the **Page URL path** field. The **{UserName}** part of the URL is the actual wildcard.

If you type ***<domain>/Members/David.aspx*** into your browser, the **Members -> Profile** page opens. The system converts the wildcard part of the URL (*David*) into a query string parameter, so that the internal URL actually looks like **<domain>/Members/Profile.aspx?username=David**. The name of the parameter is taken from the name of the wildcard, and the value is the matching part of the entered URL. The *User public profile* web part which is placed on the page recognizes the *username* parameter in the rewritten URL and displays David's profile.

Default wildcard values

You can set default values for wildcards in URLs. Use the following format in the URL path: **{<wildcard name>;<Default value>}**. Default values are useful when you want to have a wildcard in a page's URL path, but expect that the wildcard part may not always be present in the URL. The default value ensures that the system always generates the URL with a certain value in the wildcard.

For example, open the sample Community site in the **Pages** application and select the **Members** page. Switch to the **Properties -> URLs** tab, enter */Members/{UserName;David}* into the **Page URL path** field and click **Save**. If you now access the Members page on the live site, the page displays David's profile by default.

Additionally, the current value of a wildcard is always used as the default value in all other URLs on the given page that contain the same wildcard. For example, imagine a scenario with two pages that use a wildcard in their URL path: the first set to */Profile/{UserName}* and the other to */Profile/{UserName}/Details*. When the first page is accessed through the URL **<domain>/Profile/Andy.aspx**, navigation links to the second page are automatically generated as **<domain>/Profile/Andy/Details.aspx**. The current value is used even for wildcards that have a different default value set in the URL path.

Using wildcard URLs on multi-language sites

The **Page URL path** is unique for each language version of a page. As a result, you may encounter problems when referring to a page using a wildcard URL on multi-lingual sites.

For example, on the sample **Community Starter site**, the **Members/Profile** page has the **Page URL path** set to **/Members/{UserName}**. If you create a version of this page in another language, the Page URL path is automatically changed to **/Members/{UserName}-1**. This happens because the URL path needs to be unique for every page.

Now let's presume that you have the following link leading to the page: **<domain>/Members/David.aspx**. In the original version, it works fine. But if you try to click the link in the second language version, no profile is found, because the URL in this language version is **<domain>/Members/David-1.aspx**. The modification added to the end of the URL path changes the wildcard value representing the user name, which makes it impossible to find a matching user profile.

If you want to keep such links functional in all language versions, you need to define the **/Members/{UserName}** path via the **Page aliases** section, as described in Setting page aliases. When creating the alias, select **(all)** in the **Culture** drop-down list. Before you can do this, you must erase the Page URL path value for both language versions of the page.

⚠ Keep in mind that it is not possible to specify *default* wildcard values through page aliases. An alias only makes the page accessible through a given URL, it does not enforce this URL for the page.

Dots in wildcard URLs

You may encounter problems when a string containing a dot character (".") gets into the wildcard parts of URLs. A typical example of this can be found on the **Members -> Profile** page of the sample Community Starter Site.

The page's Page URL path is **/Members/{UserName}**. For the username **jack.smith**, the profile page URL is **http://<domain>/Members/jack.smith**. Since the last part of the URL after the last dot (*smith* in this case) represents the extension, the URL returns a 404 error.

One way to prevent such problems is to add validation rules that block registration with user names containing dots. If you need to allow dots in user names and use wildcard URLs with user names at the same time, move the wildcard in the Page URL to the middle of the URL, for example: **/Members/{UserName}/Profile**