

mySCADA VERSION 7 TRANSITION GUIDE

Our team at mySCADA has made maximum effort to make transition from version 6 to version 7 smooth and without troubles. We have spent significant time in keeping the compatibility between versions and doing many tests to achieve that. However, we also wanted to bring you the best SCADA system on the market, this means we had to do some decisions concerning compatibility. When you use your project created in version 6 on version 7 all the functions should be working normal, except of two parts effects and server side scripts. If you use any of these functions in your project, please read below.

What's new:

Version 7 is major update with many new functions and improvements. Among most interesting:

- completely new graphic components library
- new component “active area” allowing the integration of any HTML 5 element
- new animations: zoom, scale, opacity and circular section
- visual themes with complete user customization of runtime environment

Using effects:

Effects from version 6 are not supported, they are replaced by new effects in version 7 with greatly enhanced functionality. If you use effects in your project, you should replace them with new effects provided in version 7.

Using server side scripts:

Server Side Scripts are based on Node.js framework. In version 7, we have made several changes to be compatible with Node.js standards. This way, you can copy Node.js project and use it directly in mySCADA system. If you use Server Side Scripts in your project, please do following steps:

1. On main.js script, please add following 2 lines at the beginning:

```
myscada= require('./myscada');  
myscada. init();
```

2. If your project uses periodic time script, please copy the script into main.js and enclose it in setInterval function, like this:

```
setInterval(function() {  
  //your timed script here!  
}, 5000);
```

3. If you use time script started at given time, please use new node module called cron. Move your timed script into main.js script and enclose it in a cron function like this:

```
var CronJob = require('cron').CronJob;  
new CronJob('* * * * *', function() {  
  //your timed script here  
}, null, true, 'America/Los_Angeles');
```

To get more information about cron, please look at www <https://github.com/ncb000gt/node-cron>

With new version of Server Side Scripts, you get a lot of new functionality to make your coding easier and faster.

- better code highlight
- online interactive help
- online error checking
- integrated debugger

We have put extended effort into version 7 and we hope, you will find mySCADA version 7 the best HMI/SCADA system. We know, that merging to new system always brings some troubles, we are here to help you. Please submit a [SUPPORT TICKET](#), we will do maximum to make the transition smooth and simple.

With Best Regards

A handwritten signature in blue ink, appearing to read "Petr Svoboda".

Petr Svoboda
CEO of mySCADA Technologies