



12d<sup>®</sup> International Conference

2016

# 12d International Innovation Awards

## Joshua Allison & Michael Bozikis 12d Synergy Environment Setup

WINNER: OVERALL



Name: Joshua Allison & Michael Bozikis

Position: Design Engineers

Company: SMEC

Name Project: Synergy Environment Setup

Client: NA

Category:

- Design & Visualisation
- Survey & Construction
- Drainage, Sewer, Utilities & Rivers
- Customisation
- 12d Synergy



*Local People. Global Experience.*



# 12d International Innovation Awards

## Description of Project:

12d Synergy used as a development environment for creating 12d macros. Housing 12d setup files including the environment configuration, User folder as well as the User Library folder.

## Description of problem faced / task undertaken:

### Synergy Development

The 12d Macro Development environment is quite limited in functionality; there is no in-built IDE to allow easy compilation and testing of macros, no CVS or RCS to allow for version control and no native editor with syntax highlighting. Over the last year SMEC have used 12d Synergy as well as custom built macros to help facilitate a seamless and collaborative development environment.

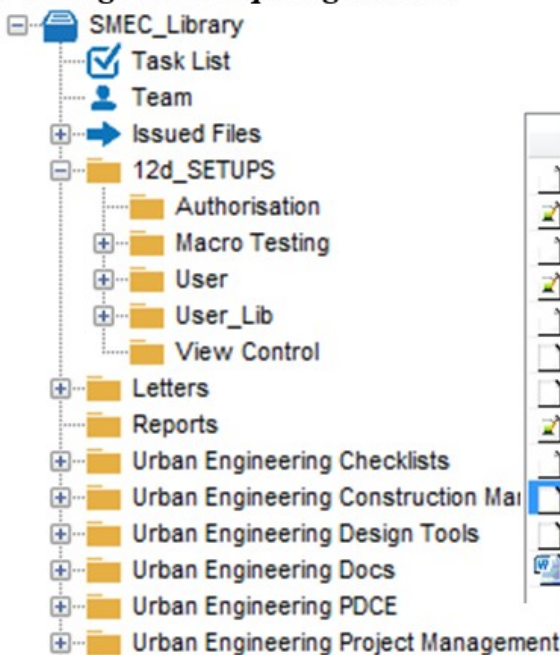
## How the problem was solved:

### Synergy to house 12d library

All of the SMEC 12d library files are housed in a Synergy Project. A batch file runs on all computers every morning to get the latest files from this library to ensure that all users are current, this also helps users work from home since they don't need to be connected to the network to access the library files.

Macros that have been written and compiled sit in a separate folder and links to those compiled \*.4do files are used in the User\_Lib folder. When a macro requires work, the file is Checked out from Synergy, and Checked back in when a functioning version of the macro is complete. This allows previous versions of macros to always be available for comparison without needing to copy and supersede files all the time. When collaborating, it also informs others as to what macros are being worked on.

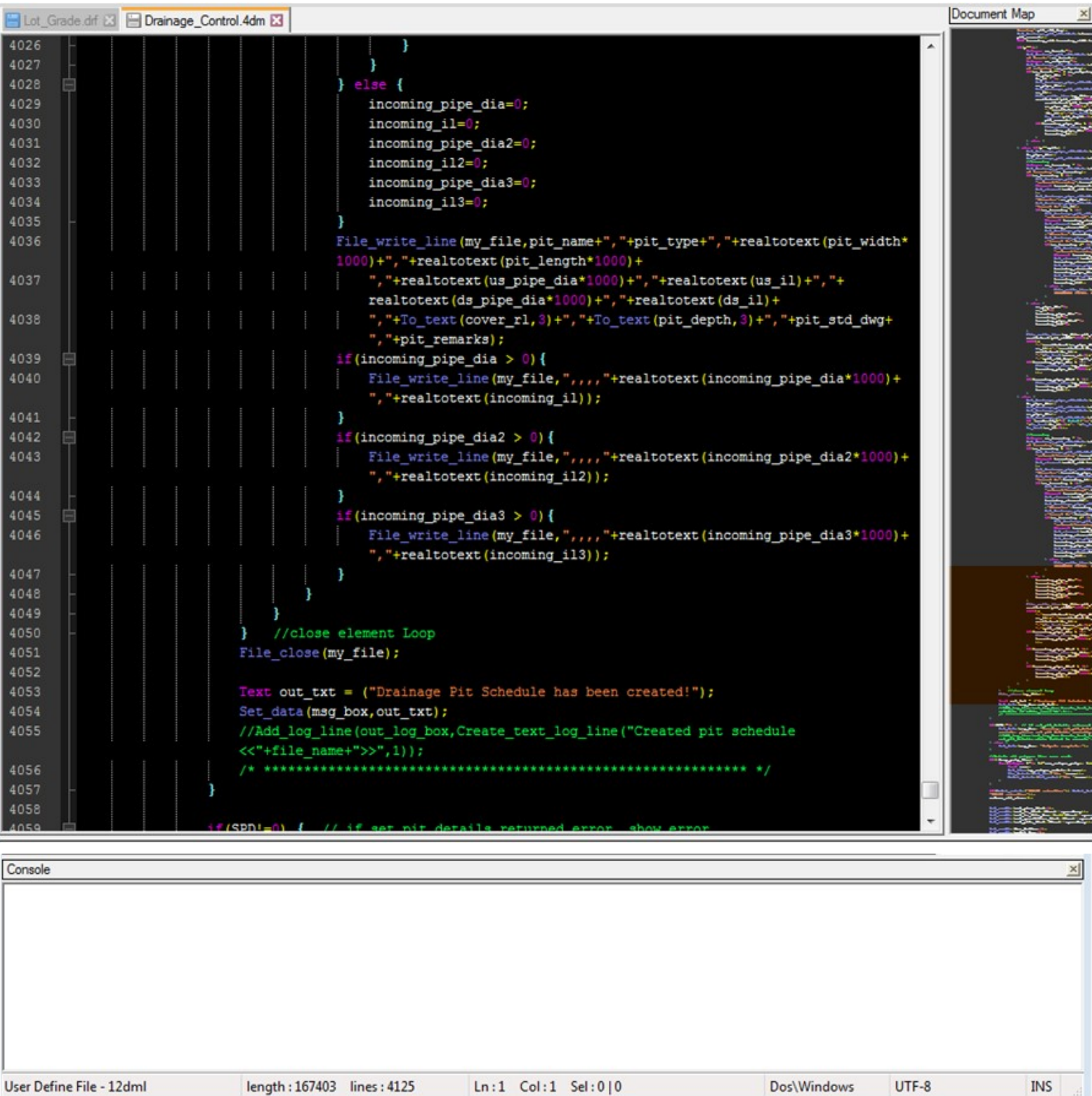
### Editing and compiling macros



File	Vers...	CheckedOut	Last Chan
Sewer_Analysis.4do	13	No	Joshua Allie
sewer_as_con.4dm	2	No	Joshua Allie
sewer_as_con.4do	2	No	Joshua Allie
Sewer_Control.4dm	12	Yes [You]	Michael Bo
Sewer_Control.4do	12	Yes [You]	Michael Bo
slf.h	3	No	Joshua Allie
smec.h	30	No	Joshua Allie
SMEC_Control.4dm	4	No	Joshua Allie
SMEC_Control.4do	4	No	Joshua Allie
smec_drains.h	18	Yes [You]	Michael Bo
sort.h	3	No	Joshua Allie
User Menu Documentation.docx	10	Yes [Joshua Allison]	Michael Bo

With advice from users on the 12d Forums, all macros are written and compiled in Notepad++. Notepad++ allows for syntax highlighting, errors output to the console from the 12d compiler and a multitude of tools such as Compare and Function Folding to help with the design process.

# 12d International Innovation Awards



```
4026     }
4027     }
4028     } else {
4029         incoming_pipe_dia=0;
4030         incoming_il=0;
4031         incoming_pipe_dia2=0;
4032         incoming_il2=0;
4033         incoming_pipe_dia3=0;
4034         incoming_il3=0;
4035     }
4036     File_write_line(my_file,pit_name+", "+pit_type+", "+realtotext(pit_width*
4037     1000)+", "+realtotext(pit_length*1000)+
4038     ", "+realtotext(us_pipe_dia*1000)+", "+realtotext(us_il)+", "+
4039     realtotext(ds_pipe_dia*1000)+", "+realtotext(ds_il)+
4040     ", "+To_text(cover_rl,3)+", "+To_text(pit_depth,3)+", "+pit_std_dwg+
4041     ", "+pit_remarks);
4042     if(incoming_pipe_dia > 0){
4043         File_write_line(my_file,",,,"+", "+realtotext(incoming_pipe_dia*1000)+
4044         ", "+realtotext(incoming_il));
4045     }
4046     if(incoming_pipe_dia2 > 0){
4047         File_write_line(my_file,",,,"+", "+realtotext(incoming_pipe_dia2*1000)+
4048         ", "+realtotext(incoming_il2));
4049     }
4050     if(incoming_pipe_dia3 > 0){
4051         File_write_line(my_file,",,,"+", "+realtotext(incoming_pipe_dia3*1000)+
4052         ", "+realtotext(incoming_il3));
4053     }
4054     }
4055     } //close element Loop
4056     File_close(my_file);
4057
4058     Text out_txt = ("Drainage Pit Schedule has been created!");
4059     Set_data(msg_box,out_txt);
4060     //Add_log_line(out_log_box,Create_text_log_line("Created pit schedule
4061     <<" +file_name+">>",1));
4062     /* ***** */
4063 }
4064
4065 #if (SPD!=0) { // if set pit details returned error show error
```

User Define File - 12dml      length : 167403    lines : 4125      Ln: 1    Col: 1    Sel: 0 | 0      Dos\Windows    UTF-8      INS

## Header files

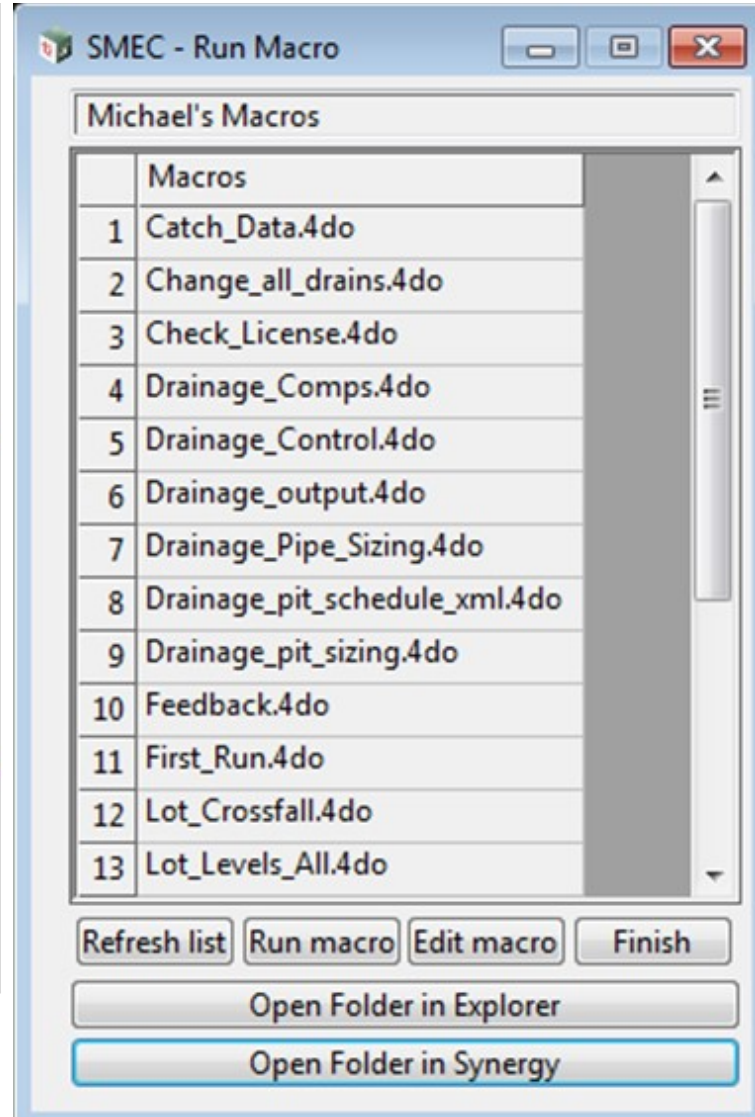
Due to the volume of functions being written and reused, SMEC library header files have been created to minimise the re-writing of functions. These header files are included in most of the macros that have been written and contain most of the commonly used functions, a sample of which is shown below.

# 12d International Innovation Awards

```

17 Real integrate(Real, Real, Real, Real, Real, Real);
18 Real integrate(Real, Real, Real, Real, Real, Real, Real);
19 Real get_area(Element, Real);
20 Real sq(Real);
21 Real absolute(Real);
22 Real high(Real, Real);
23 Real high(Element);
24 Real high(Dynamic_Element);
25 Real high(Dynamic_Real);
26 Real low(Real, Real);
27 Real low(Element);
28 Real low(Dynamic_Element);
29 Real low(Dynamic_Real);
30 Real get_wsel(Real, Real, Real, Element);
31 Real get_flow(Real, Real, Real, Real, Real);
32 Real get_wetted_perimeter(Element, Real);
33 Real get_segment_length(Real, Real, Real, Real, Real, Real, Real);
34 Real get_segment_length_top(Real, Real, Real, Real, Real, Real, Real, Real);
35 Real get_total_flow(Element);
36 Text process_text(Integer);
37 Text processing(Text, Integer, Integer);
38 Integer get_grid_edit_row(Text);
39 Integer get_grid_edit_col(Text);
40 Integer get_grid_edit_keystroke(Text);
41 void clean_model(Model);
42 void clean_model(Text);
43 Text get_cl_vertex_name(Element);
44 Dynamic_Element remove_duplicate_chainages(Dynamic_Element);
45 void model_name_parse(Text&);
46 Integer get_flow_hydro(Real, Real, Real, Real, Real&, Real&);
47 Integer get_hydro(Real, Real, Real, Element, Real&, Real&, Real&, R
48 void get_depths(Element, Real, Real, Real&, Real&);
49 void redraw_views(Model);
50 void redraw_all_views(void);
51 Text realltotext(Real);
52 void pt(Text);
53 void pt(Integer);
54 Text tt(Integer);
55 void pt(Real);

```



## *Modifying macros directly from 12d*

One issue with creating new macros or testing old macros that are being updated is linking them in 12d somehow without having to constantly going to Utilities > Macros > Run > Run, searching for the macro and then running it, only to have it not work as expected, recompiling and again going through that process.

To help expedite the testing, a tool inside 12d has been developed to allow developers to test and modify macros that have been placed into the Synergy "Macro Testing" folder. After a macro has been written/modified and recompiled, the user simply needs to find the modified macro from the panel and hit "Run macro". "Edit macro" will open the \*.4dm in the default text editor and if required, hitting "Open Folder in Synergy" will open the correct Project in Synergy and find the testing folder.