AUTODESK UNIVERSITY 2014

A Navisworks Template to Clash Them All

Andrew Brahney – Ramboll UK

Code CM6974-L

Learning Objectives

At the end of this class, you will be able to:

About the Speaker

Andy Brahney is the northern lead Building Information Modeling (BIM) coordinator for Ramboll UK, an international, multidisciplinary design consultancy. Based in the Manchester office, Andy is the office Revit software champion, providing BIM coordination services, training, and support to offices around the company. He has 7 years of experience in the architecture, engineering, and construction industry with over 6 years of Revit experience. Andy is a committee member of the Manchester Revit User Group (MRUG), and he produced best-practice documents in the use of Navisworks project review software for animation and phasing for Revit software for Ramboll UK. He has recently finished studying for his final year BSc (Hons) in civil engineering at the University of Bolton.

Contact Information

Email: andy.brahney@ramboll.co.uk

Twitter: @Andyb431

LinkedIn: http://uk.linkedin.com/pub/andrew-brahney/27/b51/750/

Introduction

Firstly, thank you very much for choosing to attend this hands-on lab. This class is very much the product of a collaborative industry in that it brings together many of the tips and from various forums, blogs and AU classes, together with a few things I have picked up along the way in Navisworks itself.

Why?

As BIM adoption advances, there is an ever increasing need to carry out clash detection to satisfy BIM Execution Plans, produce better coordinated models and to do this in an efficient manner. This process often involves dealing with multiple file formats which originate from various software packages. Additionally, many staff are not Navisworks "experts" and setting up search sets and clash tests can be a time consuming process. The solution demonstrated in this lab is to create a Navisworks template which contains pre-defined sets of the above, together with a standardised clash detection workflow, resulting in a less user intensive, and more efficient delivery method.

Pre-requisites

This class is aimed at BIM Managers, BIM Coordinators and those in a position to implement new business processes.

Software

The following software is required to setup and use the template:

- Navisworks Manage (available as part of the Building Design Ultimate Suite)
- Navisworks Switchback (subscription app)
- Dynaworks (freeware, Dynamo: http://dyn-builds-data.s3-us-west-2.amazonaws.com/DynamoInstall0.7.2.exe, Navisworks module: http://dynamopackages.com/download/5449ac30cc26bfdc5a0000e9/1.0.0)
- Notepad C++ (freeware, available from http://notepad-plus-plus.org/)

Background Setup

Before we begin loading in models and setting up the template, we first need to do some housekeeping in how Navisworks treats the loaded files.

Navisworks File Reader

1. With this process we shall be using Navisworks to import Revit files on-the-fly, therefore we need to setup the file reader to ensure that the correct information is imported.

Start Navisworks and open the options which can be found by clicking the big N and located in the bottom right corner of the menu as shown below:

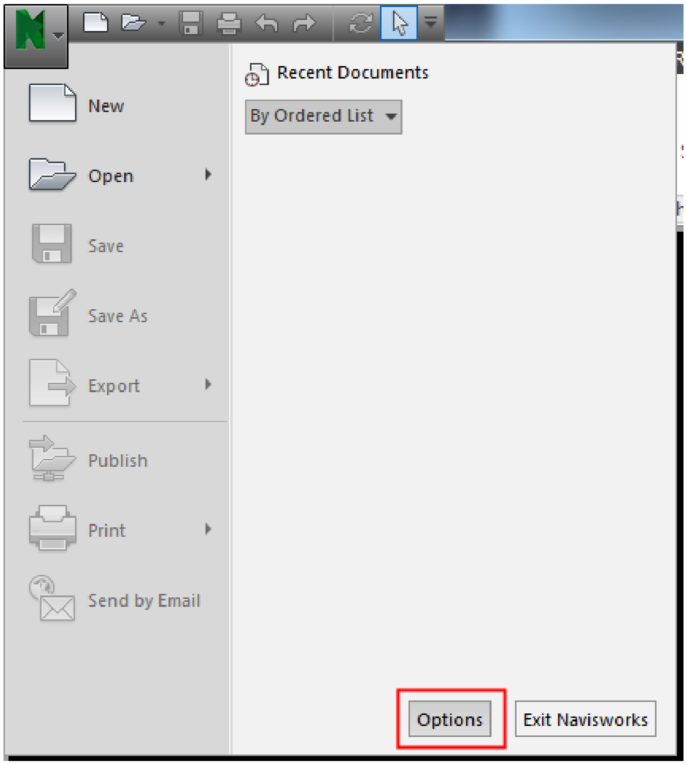
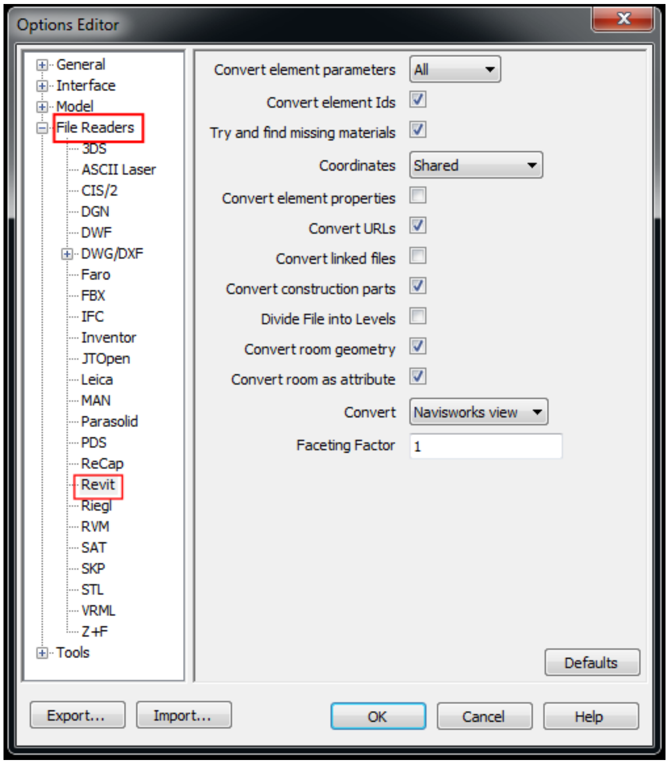


Fig. 1.1 – Navisworks File Menu

We are then presented with the following menu. We are interested in the File Readers options, so click the + sign next to this to drop down the options for the various file formats. This area allows us to tell Navisworks how to treat files, for instance, we can specify that all Revit models should be split by its levels and what parameters we would like to pull through to the model:



Convert element parameters: All
Convert element Ids ☑
Try and find missing materials ☑
Coordinates: Shared
Convert element properties ☐
Convert URLs ☑
Convert linked files ☐
Convert construction parts ☑
Divide File into Levels ☐
Convert room geometry ☑
Convert room as attribute ☑
Convert: Navisworks view
Faceting Factor: 1

Fig. 1.2 – File Reader Options for Revit

Shown above are the settings that I use and have been tested with the template; the key elements are the ‘Coordinates’, ‘Convert’ and ‘Convert Construction Parts’ properties which should be set as above. However, the other settings can be modified to suit your requirements and should work without any problems.

Revit View Setup

2. In the previous step, we specified that the ‘Convert’ should look at the Navisworks view. When importing, the file reader will look for a 3D view in the Revit file that contains with ‘Navis’. ALL geometry in this view, including scope boxes, etc. will be converted into the NWC; so in our view template at Ramboll we have turned off annotation categories.

Dynaworks Setup

3. For instructions on how to setup Dynaworks, head over to What Revit Wants which offers a great walkthrough HERE.

Create Multiple Search Sets using an XML Editor

The first step is to create the search sets which shall be used by the clash tests to select the elements to compare. We begin by loading in a ‘dummy' project, which is available in the datasets folder.

4. Import the dummy project using the Append button on the Home tab on the ribbon. Then select the ‘Navisworks Dummy Project.rvt’ file in the datasets.
5. Press the ‘Find Items’ button on the ribbon:

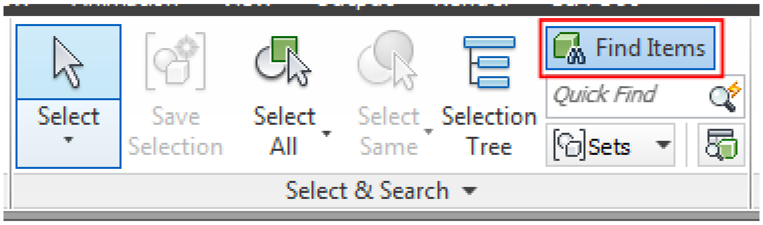
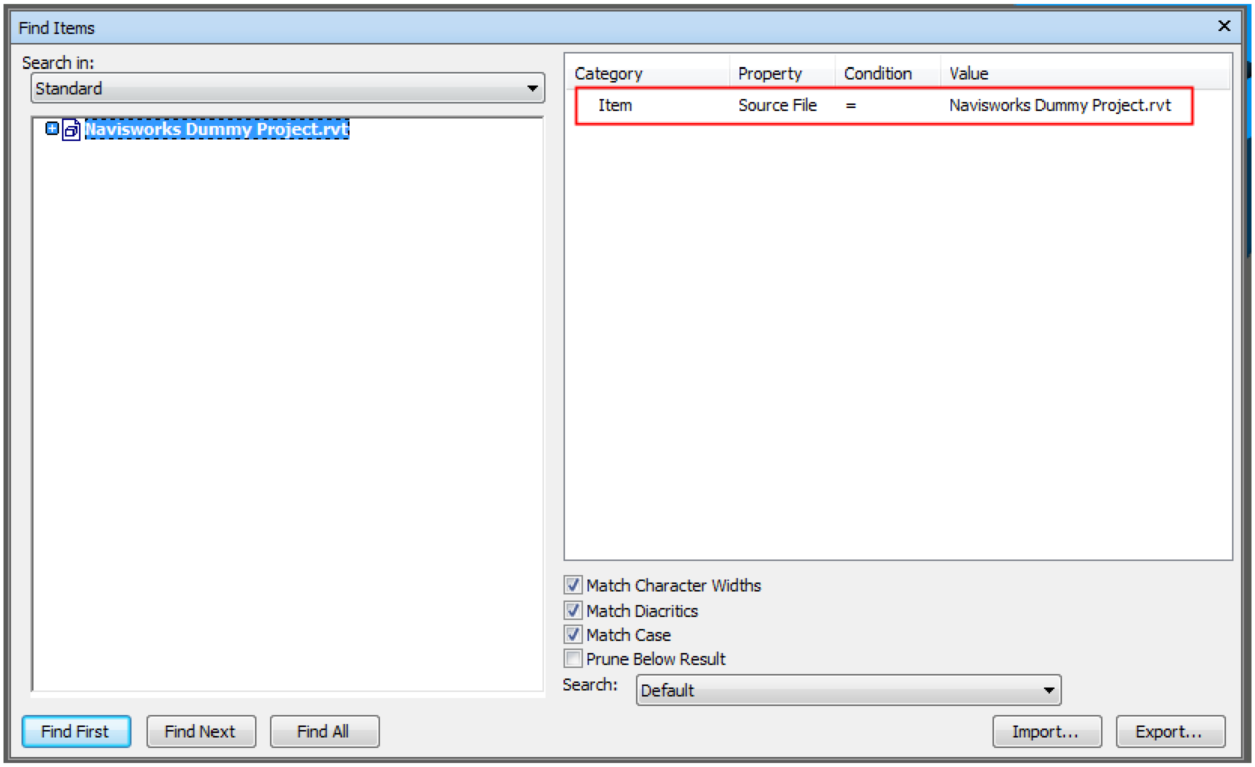


Fig. 1.3 – Select & Search Toolbar

6. Define the file to search for the elements:



Search in: Standard

Navisworks Dummy Project.rvt

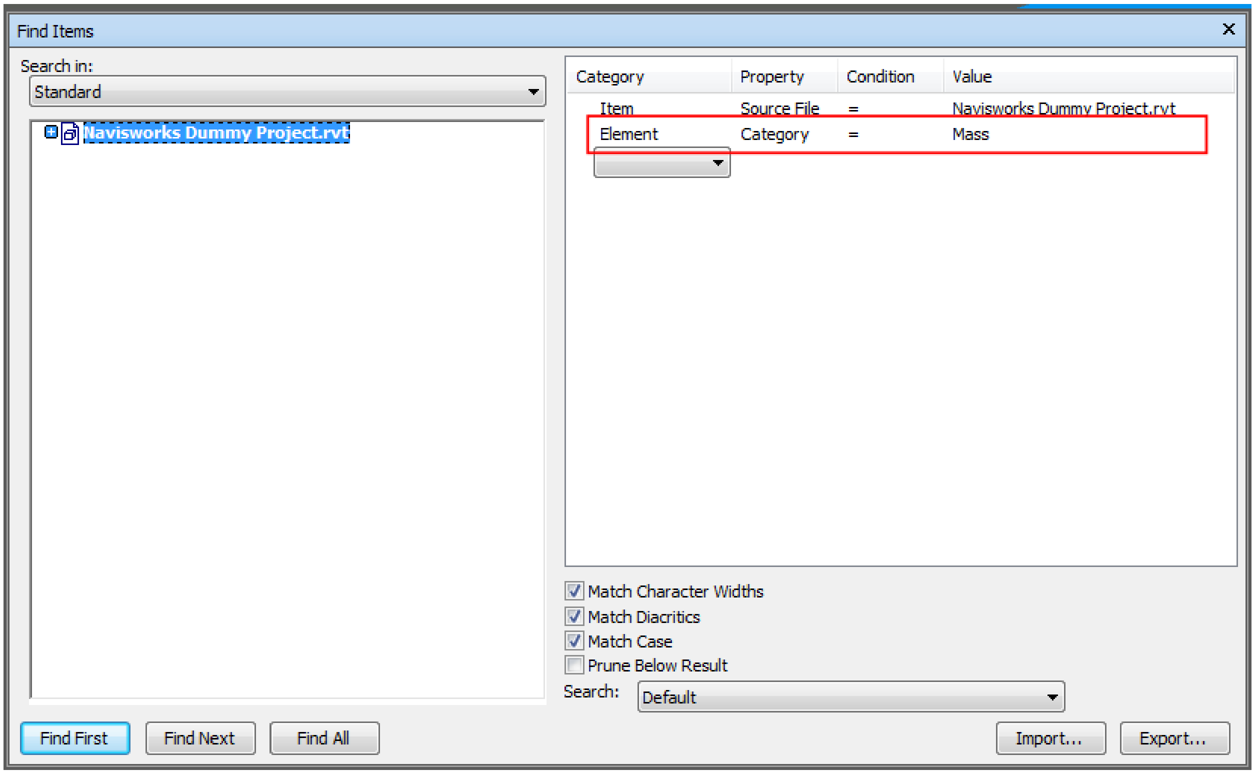
Category	Property	Condition	Value
Item	Source File	=	Navisworks Dummy Project.rvt

☑ Match Character Widths
☑ Match Diacritics
☑ Match Case
☐ Prune Below Result

Search: Default

Fig. 1.4 – Search Set Model Definition

7. Define the element category to search for:



Search in: Standard

Category	Property	Condition	Value
Item	Source File	=	Navisworks Dummy Project.rvt
Element	Category	=	Mass

☑ Match Character Widths
☑ Match Diacritics
☑ Match Case
☐ Prune Below Result

Search: Default

Fig. 1.5 – Search Set Category Definition

8. Select ‘Find All’ in the bottom left of the dialog.
9. Activate the ‘Manage Sets’ dialog:

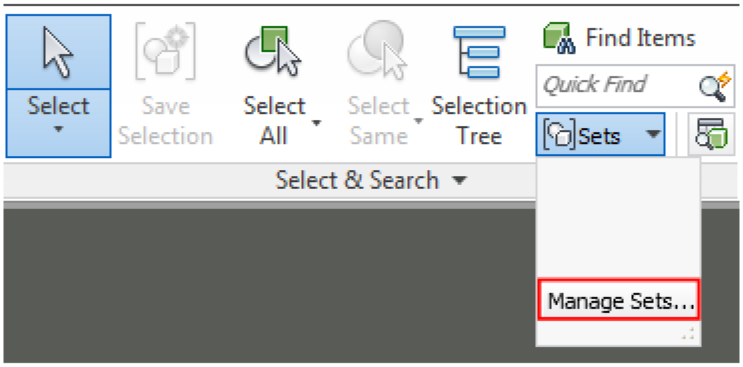


Fig. 1.6 – Set Manager Activation

10. When presented with the dialog, select the Save Search button and name the set ‘Mass’:

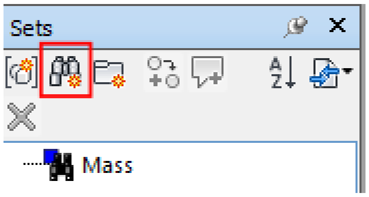


Fig. 1.7 – Save Search Button

11. Export the Search Set to XML, and save to desktop:

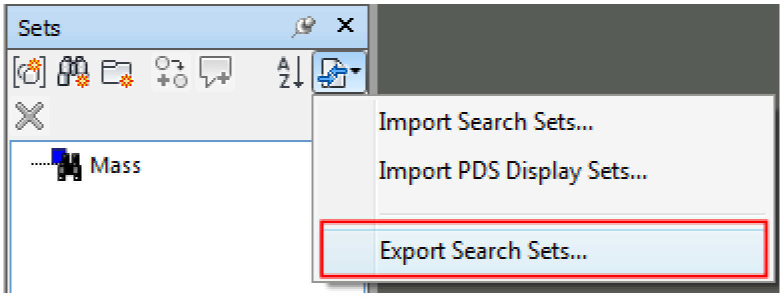


Fig. 1.8 – Export Search Sets to XML

12. Open the XML by right-clicking and selecting Edit with Notepad++:

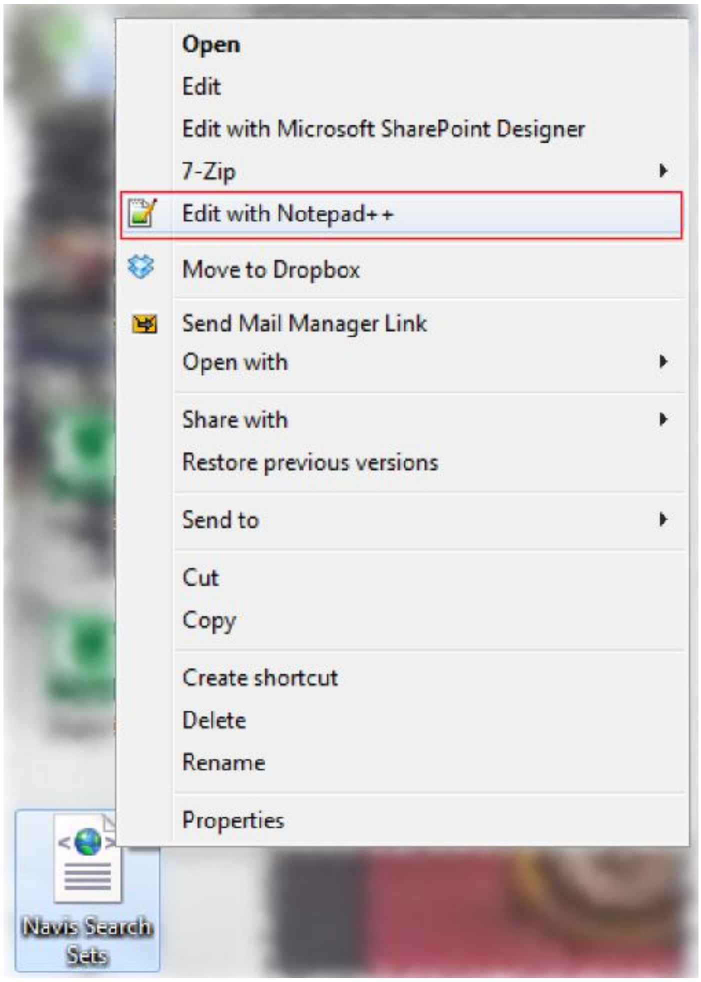
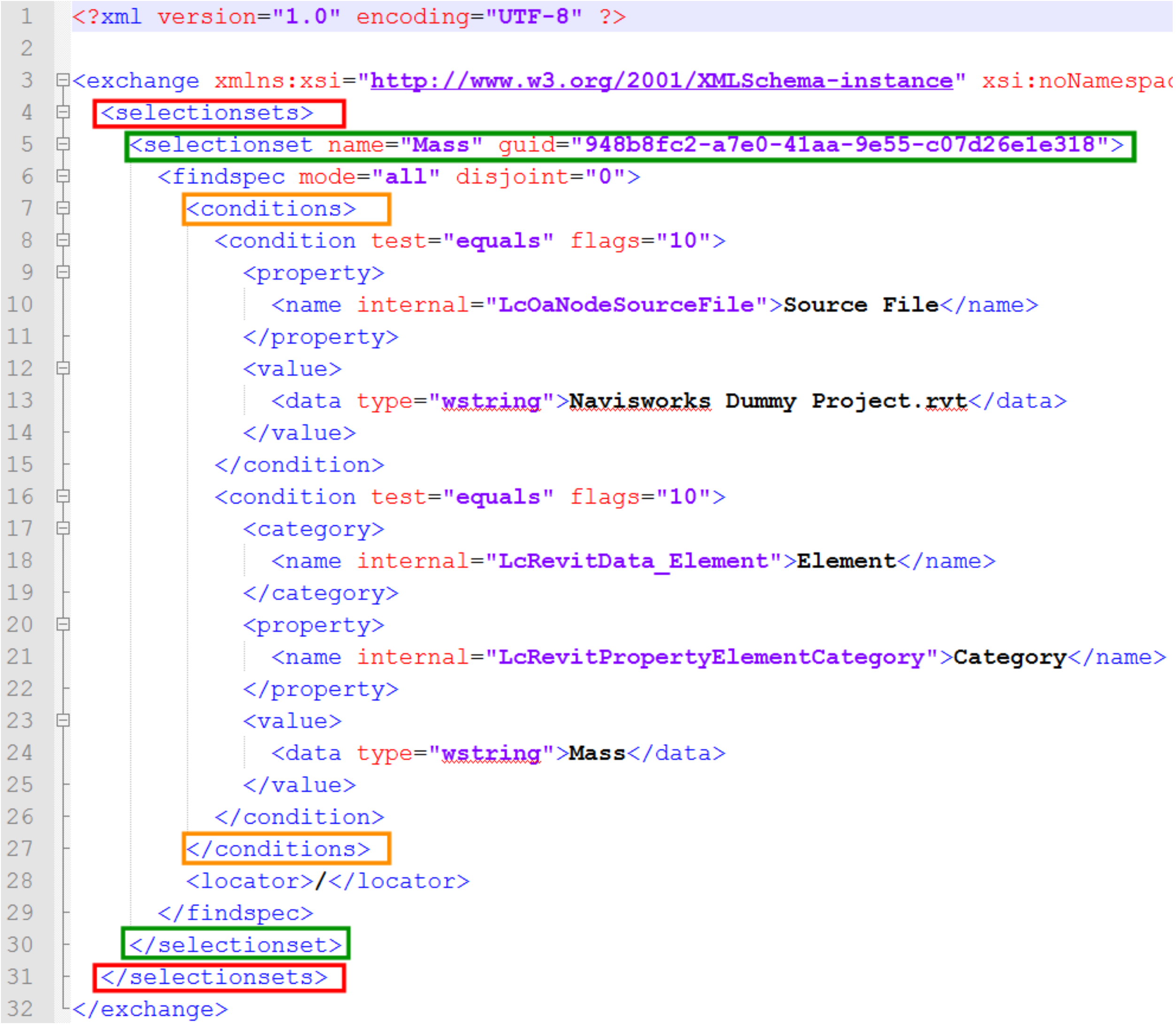


Fig.1.9 – XML Right-click Options

The structure of the XML is broken down below (I have focused in on the XML to show greater detail):



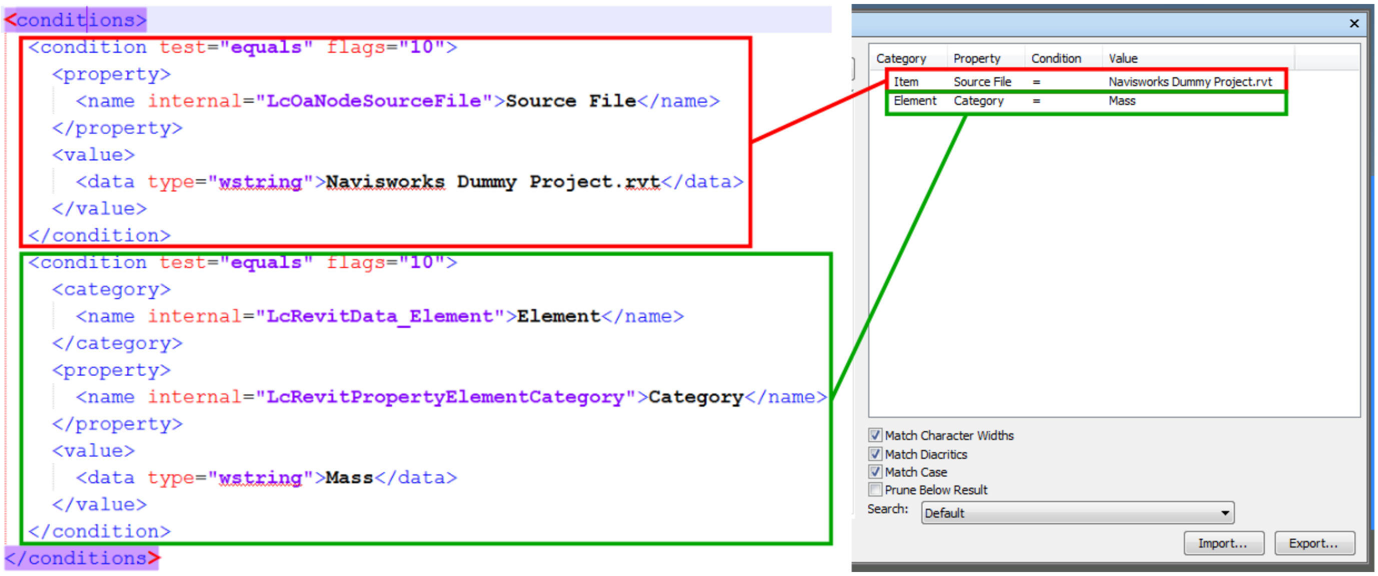
```xml
<?xml version="1.0" encoding="UTF-8" ?>

<exchange xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespac
  <selectionsets>
    <selectionset name="Mass" guid="948b8fc2-a7e0-41aa-9e55-c07d26e1e318">
      <findspec mode="all" disjoint="0">
        <conditions>
          <condition test="equals" flags="10">
            <property>
              <name internal="LcOaNodeSourceFile">Source File</name>
            </property>
            <value>
              <data type="wstring">Navisworks Dummy Project.rvt</data>
            </value>
          </condition>
          <condition test="equals" flags="10">
            <category>
              <name internal="LcRevitData_Element">Element</name>
            </category>
            <property>
              <name internal="LcRevitPropertyElementCategory">Category</name>
            </property>
            <value>
              <data type="wstring">Mass</data>
            </value>
          </condition>
        </conditions>
        <locator>/</locator>
      </findspec>
    </selectionset>
  </selectionsets>
</exchange>
```

Fig. 1.10 – Search Set XML Structure

The first thing to notice is that the XML resembles a folder structure in windows explorer with the high level red boxes show the start and end of the entire search set structure, even if there are multiple search sets. The green boxes bound the individual search set and contain the name, conditions and properties to be tested. Finally, the orange boxes shown the property conditions searched for in that particular search set; in this case the Source File and Element Category. To assist you in identifying the start and end of each section, when you click on one of the above, Notepad++ will highlight both for you.

The conditions for the search set shown below reflect the organisation in Navisworks:



```
<conditions>
  <condition test="equals" flags="10">
    <property>
      <name internal="LcOaNodeSourceFile">Source File</name>
    </property>
    <value>
      <data type="wstring">Navisworks Dummy Project.rvt</data>
    </value>
  </condition>
  <condition test="equals" flags="10">
    <category>
      <name internal="LcRevitData_Element">Element</name>
    </category>
    <property>
      <name internal="LcRevitPropertyElementCategory">Category</name>
    </property>
    <value>
      <data type="wstring">Mass</data>
    </value>
  </condition>
</conditions>
```

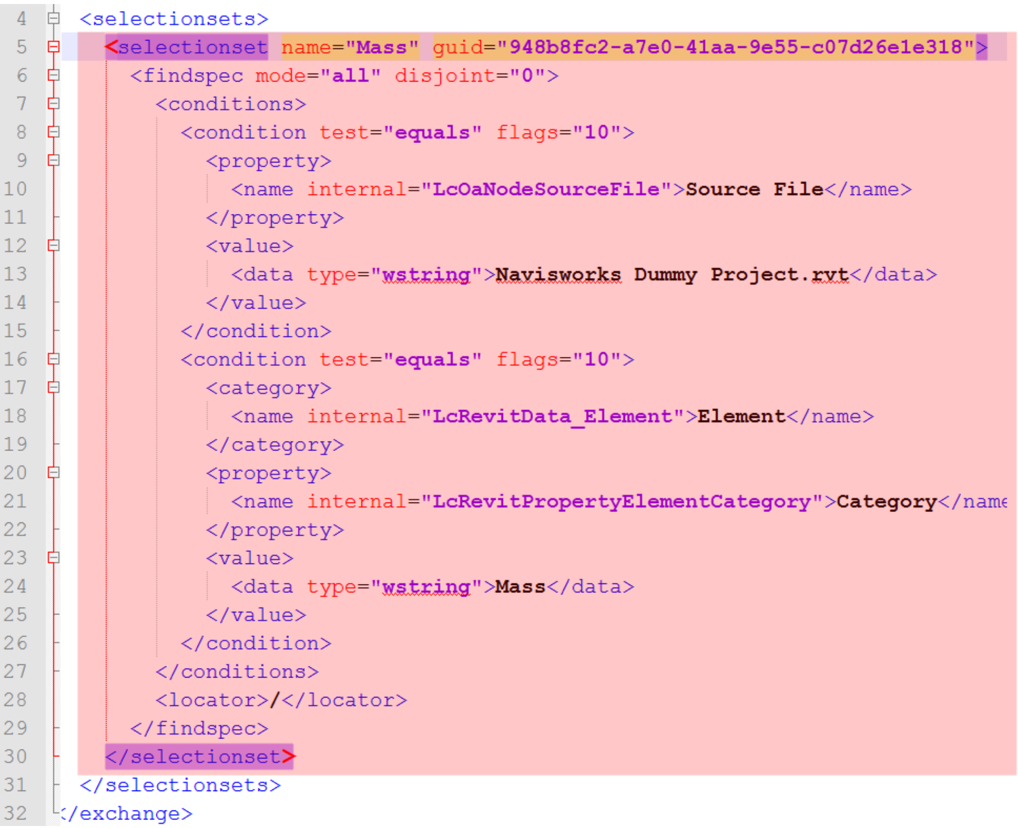
Category	Property	Condition	Value
Item	Source File	=	Navisworks Dummy Project.rvt
Element	Category	=	Mass

☑ Match Character Widths
☑ Match Diacritics
☑ Match Case
☐ Prune Below Result
Search: Default

Fig. 1.11 – Comparison of Navisworks Search Options & XML Export

Now that's enough about the XML structure, let's get back to the template!

13. Copy the section of the XML highlighted below:



```
 <selectionsets>
  <selectionset name="Mass" guid="948b8fc2-a7e0-41aa-9e55-c07d26e1e318">
   <findspec mode="all" disjoint="0">
    <conditions>
     <condition test="equals" flags="10">
      <property>
       <name internal="LcOaNodeSourceFile">Source File</name>
      </property>
      <value>
       <data type="wstring">Navisworks Dummy Project.rvt</data>
      </value>
     </condition>
     <condition test="equals" flags="10">
      <category>
       <name internal="LcRevitData_Element">Element</name>
      </category>
      <property>
       <name internal="LcRevitPropertyElementCategory">Category</name
      </property>
      <value>
       <data type="wstring">Mass</data>
      </value>
     </condition>
    </conditions>
    <locator>/</locator>
   </findspec>
  </selectionset>
 </selectionsets>
</exchange>
```

Fig. 1.12 – Selection to be Copied

14. Move the cursor to the end of "</selectionset>" and press return to start a new line.
15. Paste the section back in to the XML:
16. Press Ctrl+F to bring up the Find dialog, then press the Replace tab. Now simply replace 'Mass' with whichever category you wish to find.
17. Repeat steps 13-16 for as many search sets required.
18. Go back into Navisworks and import the search sets we just created:

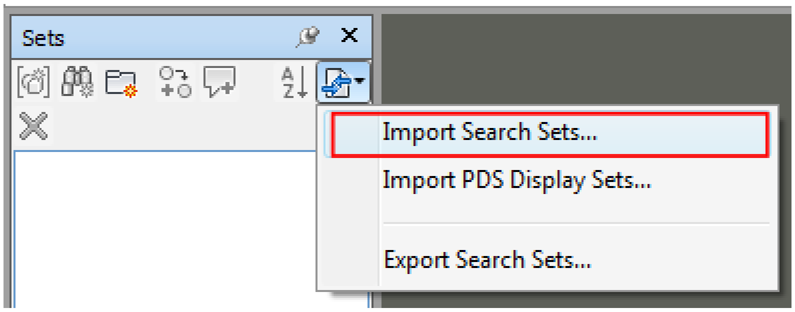


Fig. 1.13 – Importing Search Sets

19. Re-export the search sets as per step 11. This is due to the fact that when we copied the search sets, we also copied the GUID of the first set. Therefore we re-export the XML and Navisworks will assign a new GUID for each of the search sets.

And that's it; you now have yourself a base set of search sets setup! You can use this approach to setup search sets to look for any property of the imported data, such as TeklaCommon to break down the Clashing of Tekla elements with other models.

Create Multiple Clash Tests using an XML Editor

Now that the search sets have been developed, it is now time to create the clash tests.

20. Open the Clash Detective from the Tool bar on the Ribbon:

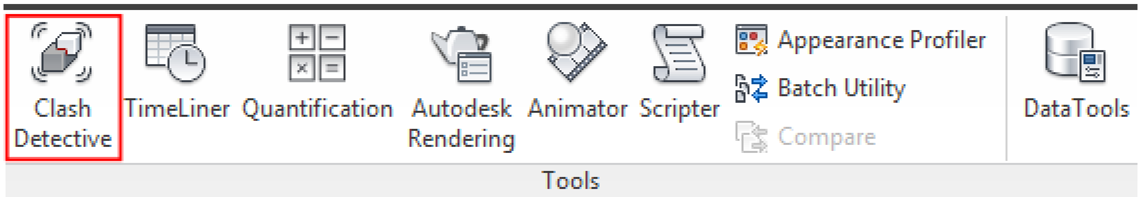
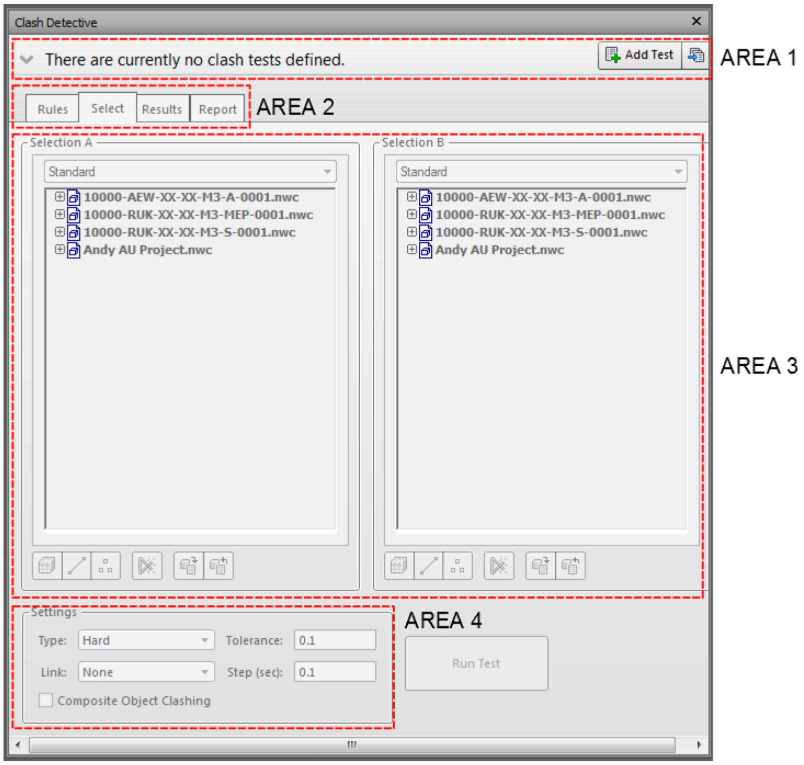


Fig. 2.1 – Clash Detective Button on Tool Tab

You will then be presented with the clash detective window as shown below:



There are currently no clash tests defined.

Selection A: Standard
- 10000-AEW-XX-XX-M3-A-0001.nwc
- 10000-RUK-XX-XX-M3-MEP-0001.nwc
- 10000-RUK-XX-XX-M3-S-0001.nwc
- Andy AU Project.nwc

Selection B: Standard
- 10000-AEW-XX-XX-M3-A-0001.nwc
- 10000-RUK-XX-XX-M3-MEP-0001.nwc
- 10000-RUK-XX-XX-M3-S-0001.nwc
- Andy AU Project.nwc

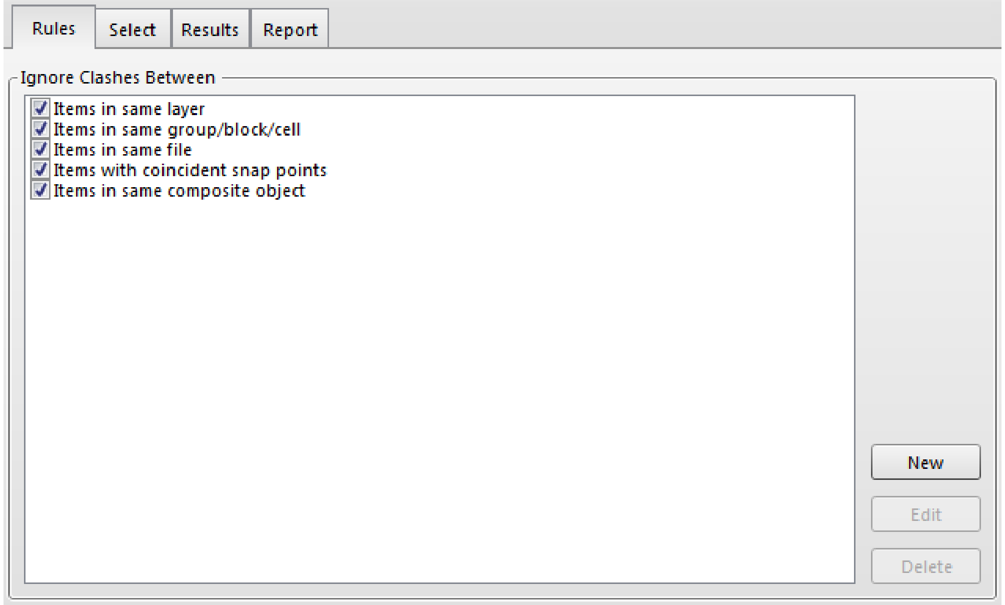
Type: Hard
Tolerance: 0.1
Link: None
Step (sec): 0.1
□ Composite Object Clashing

Fig. 2.2 – Clash Detective Dialog

Area 1 – Will display the loaded clash tests and associated stats.

Area 2 – The tabs contain the following information:

- Rules – Defines when a clash should be ignored, such as when items are from the same model:

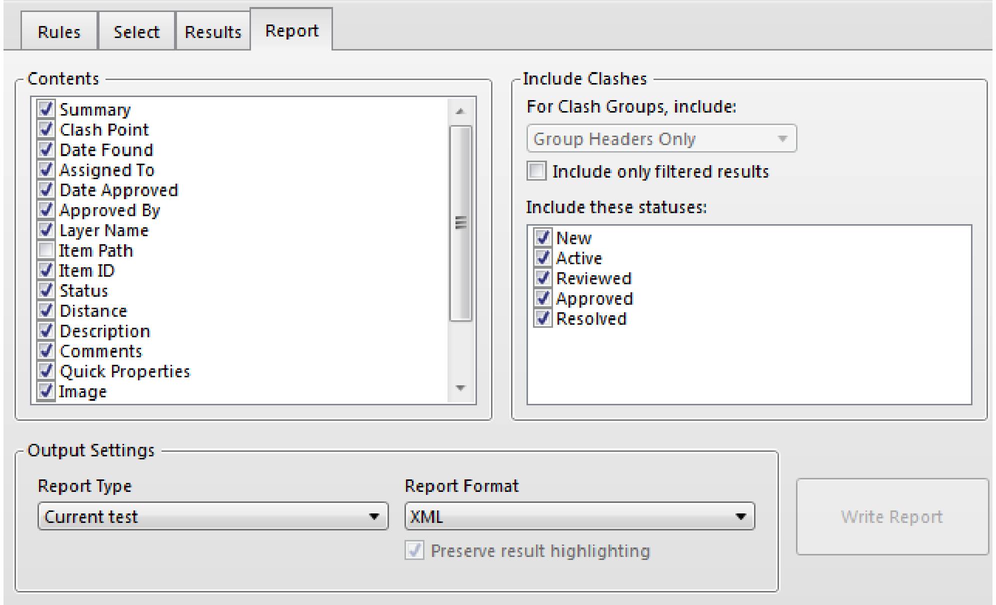


Ignore Clashes Between

- ☑ Items in same layer
- ☑ Items in same group/block/cell
- ☑ Items in same file
- ☑ Items with coincident snap points
- ☑ Items in same composite object

Fig. 2.3 – Rules Tab of Clash Detective

- Select – Displays the elements which have been selected in the clash test selected in Area 1, as shown in Fig. 1.14.
- Results – Displays the results of the clash tests.
- Report – Contains options for the export of a clash report as shown below:



Contents

- ☑ Summary
- ☑ Clash Point
- ☑ Date Found
- ☑ Assigned To
- ☑ Date Approved
- ☑ Approved By
- ☑ Layer Name
- ☐ Item Path
- ☑ Item ID
- ☑ Status
- ☑ Distance
- ☑ Description
- ☑ Comments
- ☑ Quick Properties
- ☑ Image

Include Clashes

For Clash Groups, include: Group Headers Only

☐ Include only filtered results

Include these statuses:

- ☑ New
- ☑ Active
- ☑ Reviewed
- ☑ Approved
- ☑ Resolved

Output Settings

Report Type: Current test

Report Format: XML

☑ Preserve result highlighting

Fig. 2.4 – Clash Report Export Options.

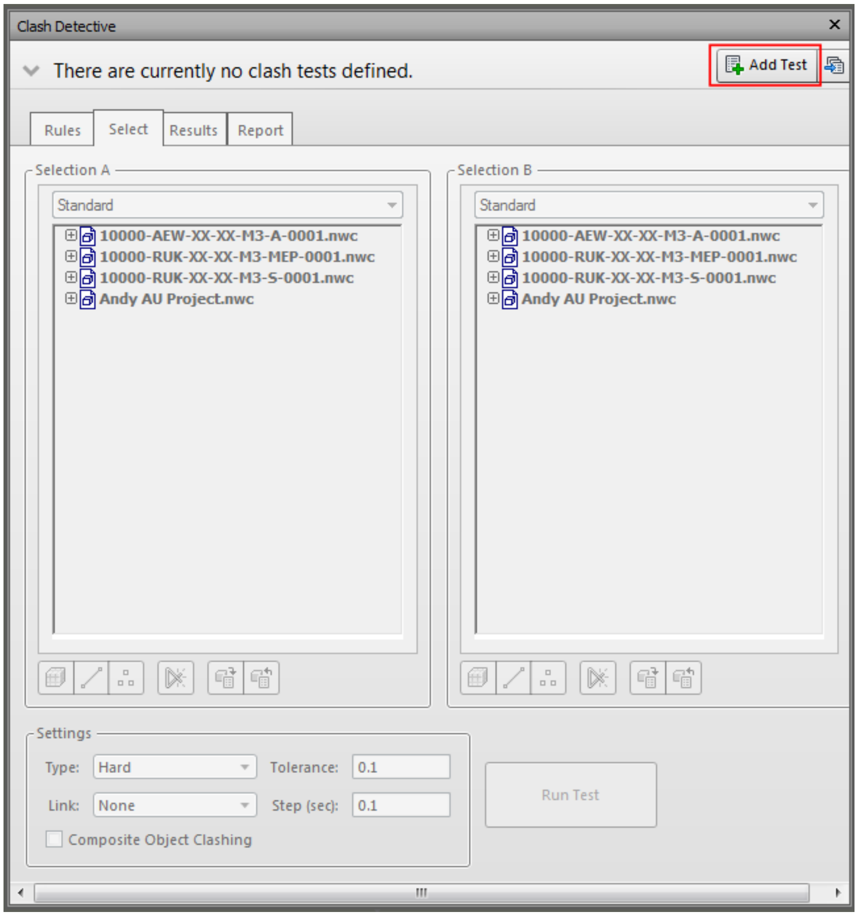
Area 3 – Displays the content of the selected tab from Area 2.
Area 4 – Displays the type of clash test to be carried out which can include:

- Hard – Where the geometry intersects each other.

- Hard (Conservative) - Two objects are treated as intersecting even though the geometry triangles do not.
- Clearance – two objects are treated as intersecting when they come within a specified distance of each other. Selecting this clash type also detects any hard clashes.
- Duplicates – Detects if two objects are identical both in type and position.

The first step, as with the search sets, is to create a clash test using the 'traditional method' inside Navisworks

21. Create a blank clash test by pressing the Add Test button shown below:



There are currently no clash tests defined.

Selection A: Standard
10000-AEW-XX-XX-M3-A-0001.nwc
10000-RUK-XX-XX-M3-MEP-0001.nwc
10000-RUK-XX-XX-M3-S-0001.nwc
Andy AU Project.nwc

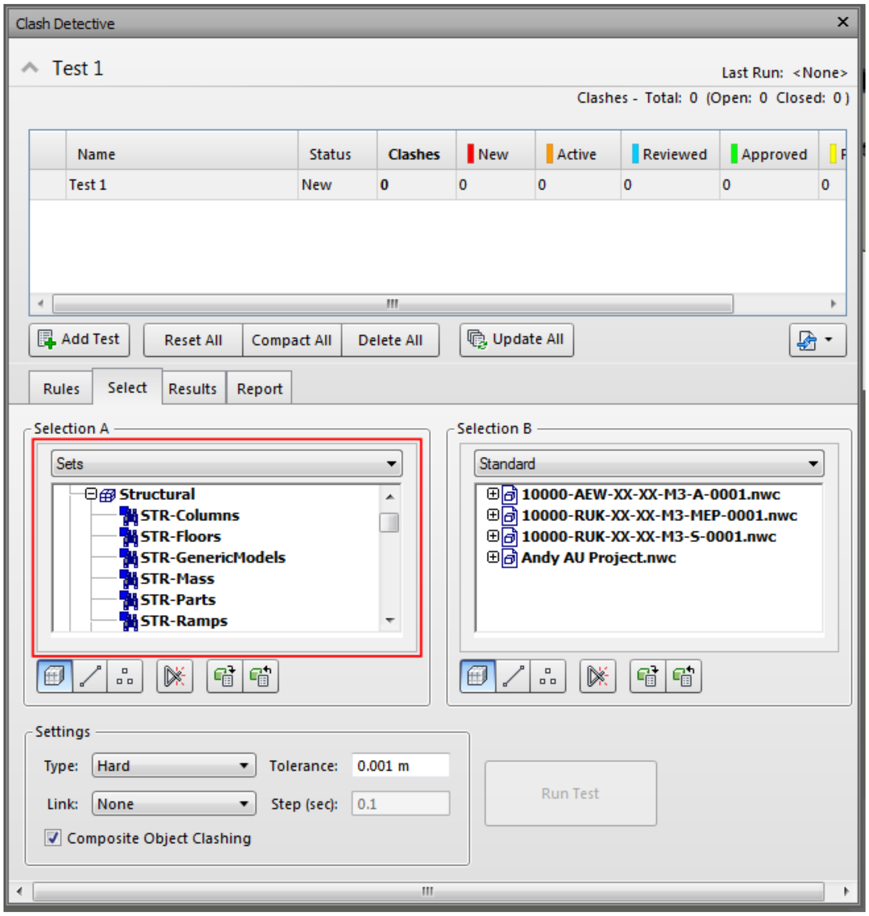
Selection B: Standard
10000-AEW-XX-XX-M3-A-0001.nwc
10000-RUK-XX-XX-M3-MEP-0001.nwc
10000-RUK-XX-XX-M3-S-0001.nwc
Andy AU Project.nwc

Type: Hard
Tolerance: 0.1
Link: None
Step (sec): 0.1
□ Composite Object Clashing

Fig. 2.5 – Add a Test to the Project

22. Dropdown the menu under selection A and select 'Sets'.

The selection A pane will now display the search sets:



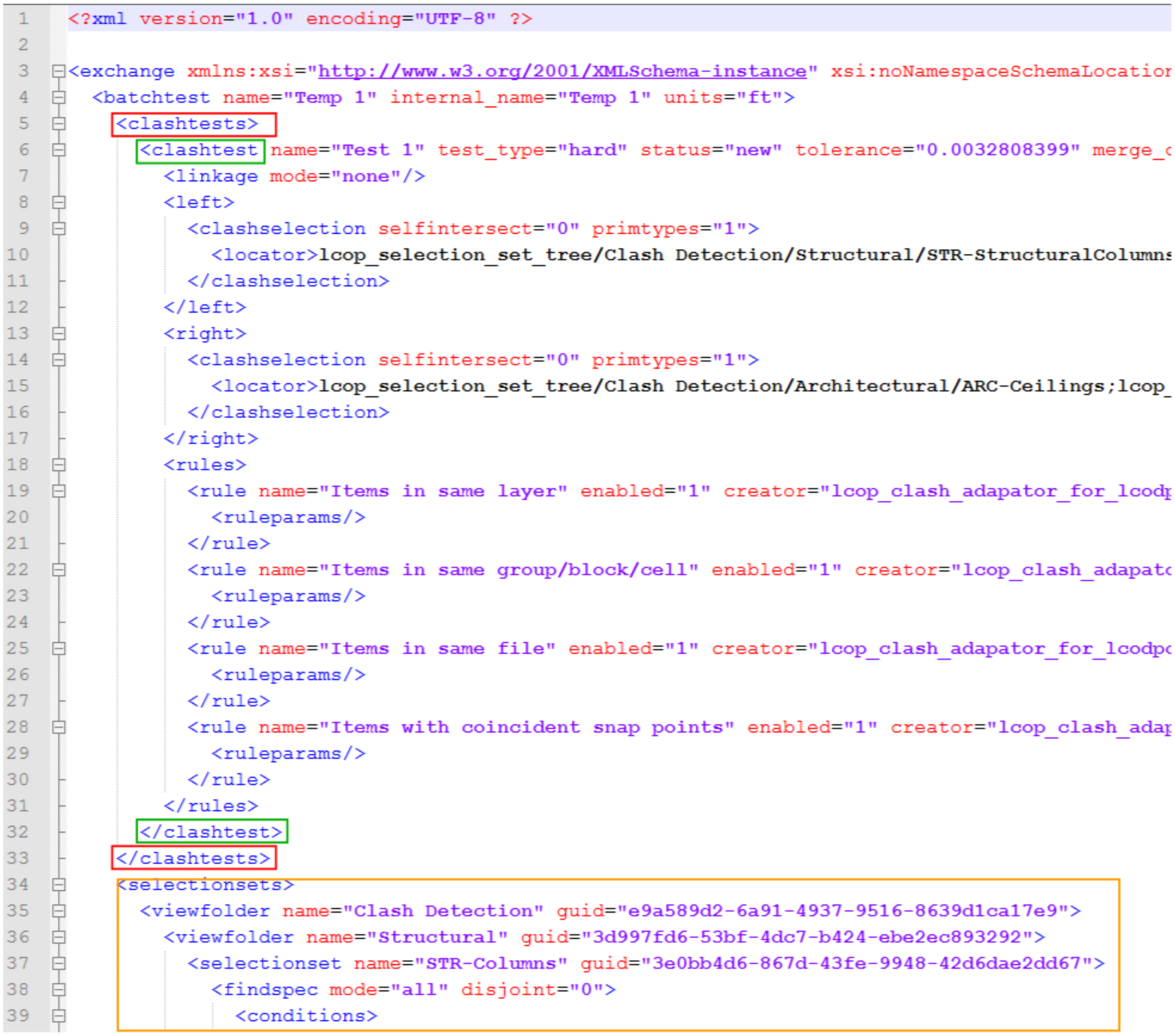
Name	Status	Clashes	New	Active	Reviewed	Approved	R
Test 1	New	0	0	0	0	0	0

Type: Hard
Tolerance: 0.001 m
Link: None
Step (sec): 0.1
☑ Composite Object Clashing

Fig. 2.6 – Search Sets Displayed in Clash Detective

For the first test, we shall create a dummy test as we did for the search set.

23. Select the Structural search sets for the Structural Columns, Structural Framing and Foundations in selection A.
24. Select the Architectural sets for Ceilings Doors, Floors, Walls & Windows in selection B.
25. Set the clash type to Hard and tolerance of 0.1m
26. Go to the rules tab and select ALL of the options.
27. Export the clash test to xml by pressing the button.
28. We are presented with an XML in a very similar format to the search sets:



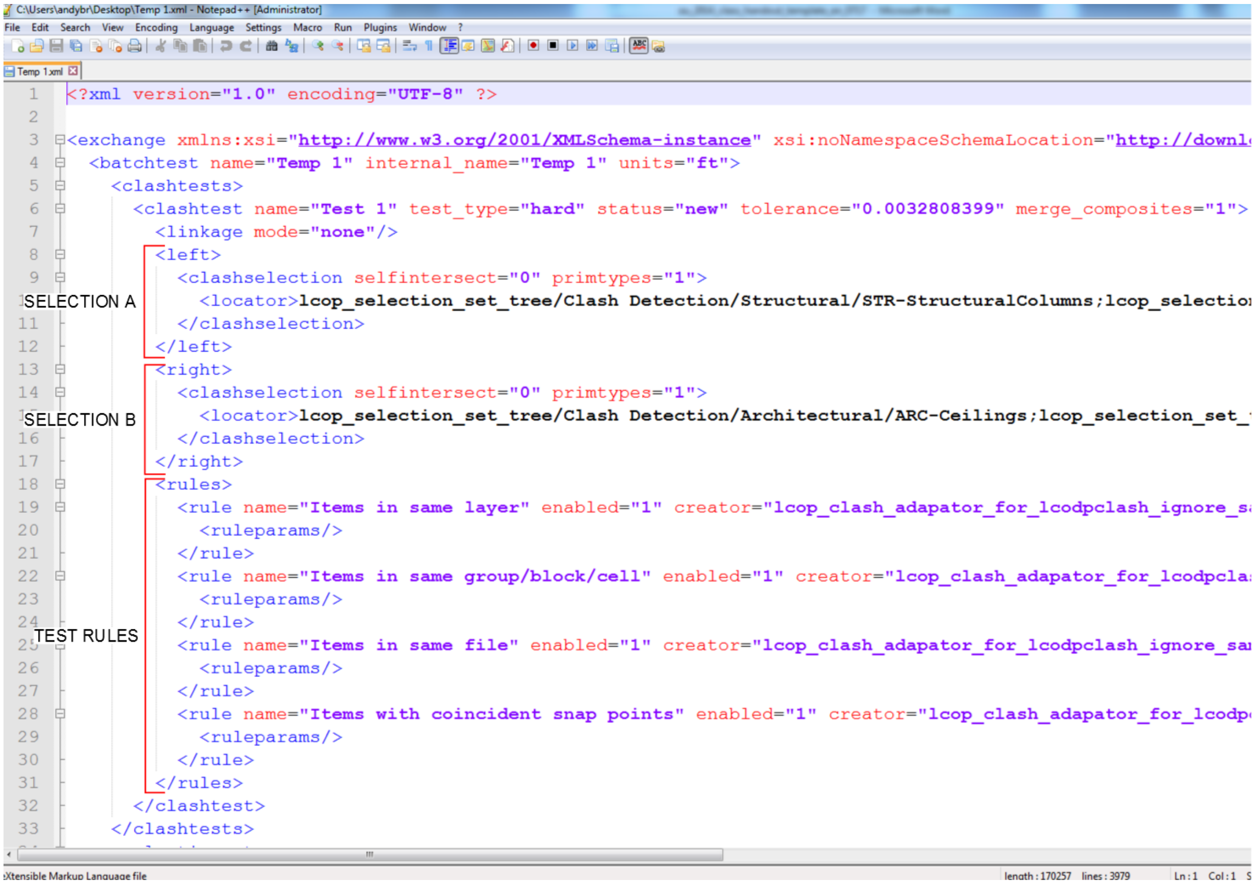
```
<?xml version="1.0" encoding="UTF-8" ?>

<exchange xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocatio
  <batchtest name="Temp 1" internal_name="Temp 1" units="ft">
    <clashtests>
      <clashtest name="Test 1" test_type="hard" status="new" tolerance="0.0032808399" merge_
        <linkage mode="none"/>
        <left>
          <clashselection selfintersect="0" primtypes="1">
            <locator>lcop_selection_set_tree/Clash Detection/Structural/STR-StructuralColumn
          </clashselection>
        </left>
        <right>
          <clashselection selfintersect="0" primtypes="1">
            <locator>lcop_selection_set_tree/Clash Detection/Architectural/ARC-Ceilings;lcop_
          </clashselection>
        </right>
        <rules>
          <rule name="Items in same layer" enabled="1" creator="lcop_clash_adapator_for_lcod
            <ruleparams/>
          </rule>
          <rule name="Items in same group/block/cell" enabled="1" creator="lcop_clash_adapat
            <ruleparams/>
          </rule>
          <rule name="Items in same file" enabled="1" creator="lcop_clash_adapator_for_lcodp
            <ruleparams/>
          </rule>
          <rule name="Items with coincident snap points" enabled="1" creator="lcop_clash_ada
            <ruleparams/>
          </rule>
        </rules>
      </clashtest>
    </clashtests>
    <selectionsets>
      <viewfolder name="Clash Detection" guid="e9a589d2-6a91-4937-9516-8639d1ca17e9">
        <viewfolder name="Structural" guid="3d997fd6-53bf-4dc7-b424-ebe2ec893292">
          <selectionset name="STR-Columns" guid="3e0bb4d6-867d-43fe-9948-42d6dae2dd67">
            <findspec mode="all" disjoint="0">
              <conditions>
```

Fig. 2.7 – Clash Test XML Structure

As can be seen above, the ‘<selectionsets>’ has simply been replaced with ‘<clashtests>’ and the same applies to the individual clash tests also. At the bottom of the XML, highlighted in orange, are the search sets which are also exported in the file. Note that the list of these will be considerably longer than that shown. The search sets are unnecessary and may simply be deleted from the file. However, it is very important that you do not delete the ‘<selectionsets>’ and ‘</selectionsets>’ parentheses as if you do, the clash test will not work when imported.

The file structure of the clash test differs considerably from that of the search set:


C:\Users\andybr\Desktop\Temp 1.xml - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
Temp 1.xml
SELECTION A
SELECTION B
TEST RULES
eXtensible Markup Language file
length : 170257 lines : 3979 Ln : 1 Col : 1

```
1  <?xml version="1.0" encoding="UTF-8" ?>
2
3  <exchange xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://downl
4    <batchtest name="Temp 1" internal_name="Temp 1" units="ft">
5      <clashtests>
6        <clashtest name="Test 1" test_type="hard" status="new" tolerance="0.0032808399" merge_composites="1">
7          <linkage mode="none"/>
8          <left>
9            <clashselection selfintersect="0" primtypes="1">
10             <locator>lcop_selection_set_tree/Clash Detection/Structural/STR-StructuralColumns;lcop_selectio
11           </clashselection>
12         </left>
13         <right>
14           <clashselection selfintersect="0" primtypes="1">
15             <locator>lcop_selection_set_tree/Clash Detection/Architectural/ARC-Ceilings;lcop_selection_set_
16           </clashselection>
17         </right>
18         <rules>
19           <rule name="Items in same layer" enabled="1" creator="lcop_clash_adapator_for_lcodpclash_ignore_s
20             <ruleparams/>
21           </rule>
22           <rule name="Items in same group/block/cell" enabled="1" creator="lcop_clash_adapator_for_lcodpcla
23             <ruleparams/>
24           </rule>
25           <rule name="Items in same file" enabled="1" creator="lcop_clash_adapator_for_lcodpclash_ignore_sa
26             <ruleparams/>
27           </rule>
28           <rule name="Items with coincident snap points" enabled="1" creator="lcop_clash_adapator_for_lcodp
29             <ruleparams/>
30           </rule>
31         </rules>
32       </clashtest>
33     </clashtests>
```

Fig. 2.8 – Clash Test Structure in Detail

Line 4 contains the name and units of the clash test. Note that although working in metric, the internal units that Navisworks uses are imperial; therefore this value shall always be ‘ft’.

Line 6 shows further detail of the clash test to be carried out. The first piece of information shows that this test is looking for a hard clash and that this is a ‘new’ test for the project, i.e. it has not yet been executed. The tolerance value we provided was 100mm, however, Navisworks has converted this to feet as per the internal units. Finally, the merge_composites value (0 being ‘No’ and 1 being ‘Yes’), is telling Navisworks whether composite objects, such as floors with finishes, should be treat as on element i.e. not clashed against each other.

Line 7 provides information about the linkage of the clash test to a timeliner sequence. This could be useful for instance, when testing a construction sequence; after all, a member could be clash free in its final state, but may clash with some temporary works during construction. This function is not covered in this class, but may be of some interest to those of you involved in the planning of construction activities.

Lines 8 – 12 show the elements we chose in Selection A. Firstly, the ‘selfintersect’ value shows that we chose to ignore self-intersecting geometry in this selection. An example of self-intersecting geometry could be a concrete encased steel column which has not had a void cut around the steel column internally. You can see that the parenthesis for this are called ‘left’, this is because Selection A is in the left hand pane in the clash detection dialog. The black text shows the addresses of the included search sets.

Lines 13 – 17 show the same information, but for Selection B.

Lines 18 – 31 display the settings for the options we specified on the Rules tab.

29. To duplicate the clash test, simply highlight everything from <clashtest> to </clashtest> and copy to clipboard.
30. Place cursor at the end of </clashtest> and press return
31. Paste the information from the clipboard.

To modify the search sets used and the name, simply overwrite the information in the relevant area.

Now you can use this approach to quickly generate multiple clash test.

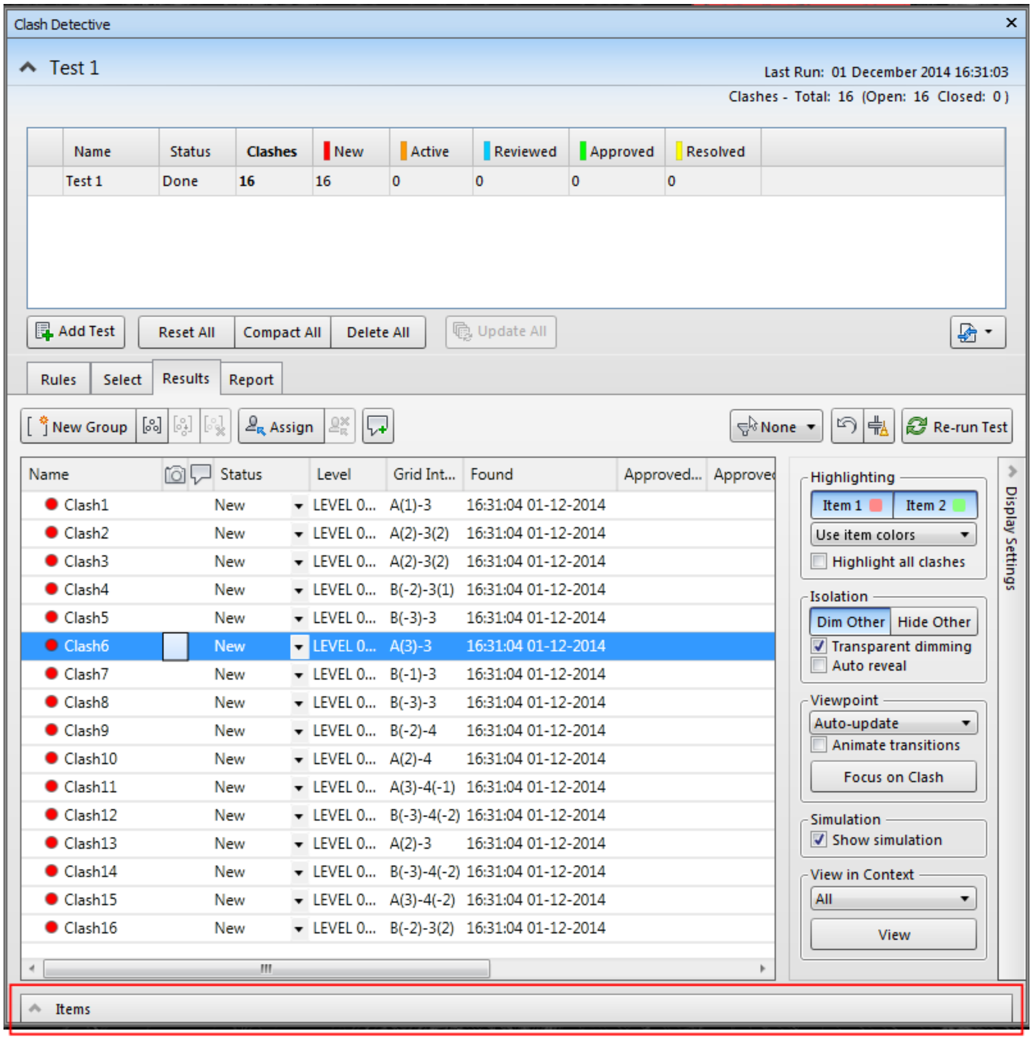
Carry Out Basic Customisation of Clash Reports

The clash report is the outcome of all of the work we have done so far. In this section we shall look at a few basic tips and tricks for the modification of the output from Navisworks.

Tip 1 – Group Clashes:

Some elements such as beams or floors may have multiple clashes associated with them, which shall be shown as individual clahses on the clash report. To ease navigation and reduce the size of the report, Navisworks can group clashes for a given item:

32. In the Clash Detective window, select the items pane, as highlighted below:



Last Run: 01 December 2014 16:31:03
Clashes - Total: 16 (Open: 16 Closed: 0)

Name	Status	Clashes	New	Active	Reviewed	Approved	Resolved
Test 1	Done	16	16	0	0	0	0

Name	Status	Level	Grid Int...	Found	Approved...	Approved
● Clash1	New	LEVEL 0...	A(1)-3	16:31:04 01-12-2014		
● Clash2	New	LEVEL 0...	A(2)-3(2)	16:31:04 01-12-2014		
● Clash3	New	LEVEL 0...	A(2)-3(2)	16:31:04 01-12-2014		
● Clash4	New	LEVEL 0...	B(-2)-3(1)	16:31:04 01-12-2014		
● Clash5	New	LEVEL 0...	B(-3)-3	16:31:04 01-12-2014		
● Clash6	New	LEVEL 0...	A(3)-3	16:31:04 01-12-2014		
● Clash7	New	LEVEL 0...	B(-1)-3	16:31:04 01-12-2014		
● Clash8	New	LEVEL 0...	B(-3)-3	16:31:04 01-12-2014		
● Clash9	New	LEVEL 0...	B(-2)-4	16:31:04 01-12-2014		
● Clash10	New	LEVEL 0...	A(2)-4	16:31:04 01-12-2014		
● Clash11	New	LEVEL 0...	A(3)-4(-1)	16:31:04 01-12-2014		
● Clash12	New	LEVEL 0...	B(-3)-4(-2)	16:31:04 01-12-2014		
● Clash13	New	LEVEL 0...	A(2)-3	16:31:04 01-12-2014		
● Clash14	New	LEVEL 0...	B(-3)-4(-2)	16:31:04 01-12-2014		
● Clash15	New	LEVEL 0...	A(3)-4(-2)	16:31:04 01-12-2014		
● Clash16	New	LEVEL 0...	B(-2)-3(2)	16:31:04 01-12-2014		

Highlighting
Item 1 🟥 Item 2 🟩
Use item colors
☐ Highlight all clashes

Isolation
Dim Other Hide Other
☑ Transparent dimming
☐ Auto reveal

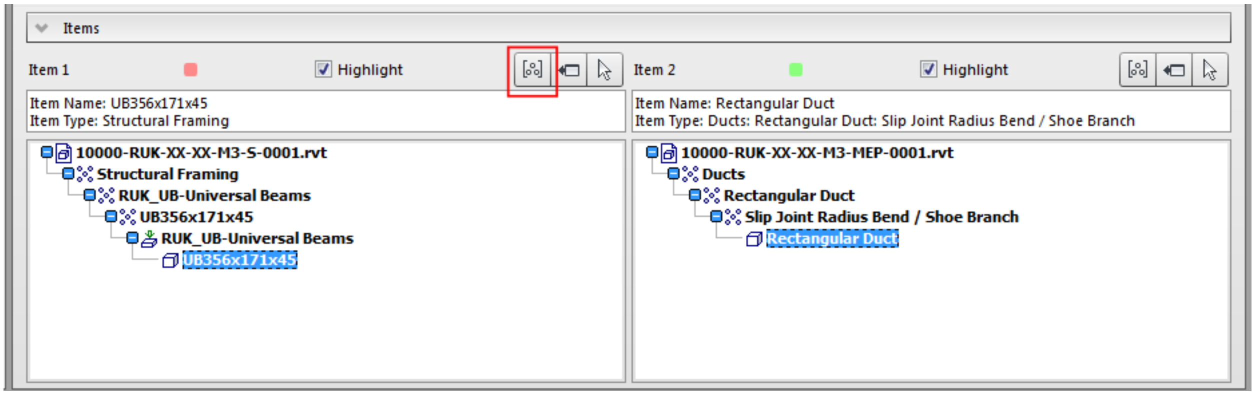
Viewpoint
Auto-update
☐ Animate transitions

Simulation
☑ Show simulation

View in Context
All

Fig. 3.1 – Items Menu of Clash Detective

33. Once expanded, press the highlighted button for the element you wish to group clashes for:



Items

Item 1 ☑ Highlight

Item Name: UB356x171x45
Item Type: Structural Framing

- 10000-RUK-XX-XX-M3-S-0001.rvt
  - Structural Framing
    - RUK_UB-Universal Beams
      - UB356x171x45
        - RUK_UB-Universal Beams
          - UB356x171x45

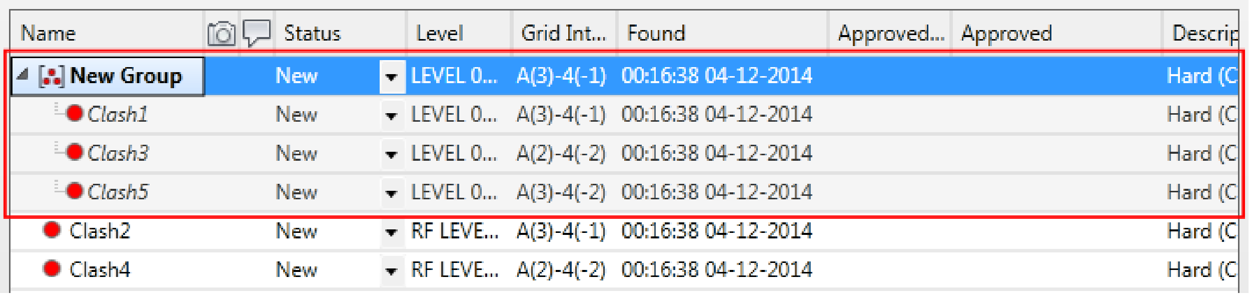
Item 2 ☑ Highlight

Item Name: Rectangular Duct
Item Type: Ducts: Rectangular Duct: Slip Joint Radius Bend / Shoe Branch

- 10000-RUK-XX-XX-M3-MEP-0001.rvt
  - Ducts
    - Rectangular Duct
      - Slip Joint Radius Bend / Shoe Branch
        - Rectangular Duct

Fig. 3.2 – Expanded Item Menu of Clash Detective

You should now see that in the main window, that the clashes have now been grouped:



Name			Status	Level	Grid Int...	Found	Approved...	Approved	Descrip
New Group			New	LEVEL 0...	A(3)-4(-1)	00:16:38 04-12-2014			Hard (C
Clash1			New	LEVEL 0...	A(3)-4(-1)	00:16:38 04-12-2014			Hard (C
Clash3			New	LEVEL 0...	A(2)-4(-2)	00:16:38 04-12-2014			Hard (C
Clash5			New	LEVEL 0...	A(3)-4(-2)	00:16:38 04-12-2014			Hard (C
Clash2			New	RF LEVE...	A(3)-4(-1)	00:16:38 04-12-2014			Hard (C
Clash4			New	RF LEVE...	A(2)-4(-2)	00:16:38 04-12-2014			Hard (C

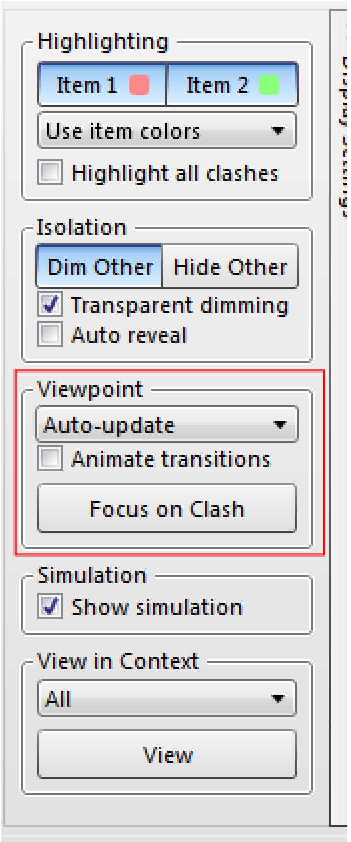
Fig. 3.3 – Grouped Clashes

Tip 2 – Export to HTML:

There are various format options for the export of the clash reports, but I feel that the most useful is the HTML file. The HTML option contains embedded images of the clashes to further communicate what the clash looks like with the team and can aid in navigation.

Tip 3 – Customise Viewpoints:

When Navisworks creates the viewpoints for the clashes, they can often be isolated and lack context, which can make the images useless. To overcome this, you can modify the views by simply ensuring that the Auto-update box is checked on the display setting of the results tab, as shown below, and then navigating around the view. Once you are happy, click on another clash and the view shall be saved.



Highlighting

Item 1 🟥 | Item 2 🟩

Use item colors

□ Highlight all clashes

Isolation

Dim Other | Hide Other

☑ Transparent dimming

□ Auto reveal

Viewpoint

Auto-update

□ Animate transitions

Focus on Clash

Simulation

☑ Show simulation

View in Context

All

View

Fig. 3.2 – Display Settings Panel

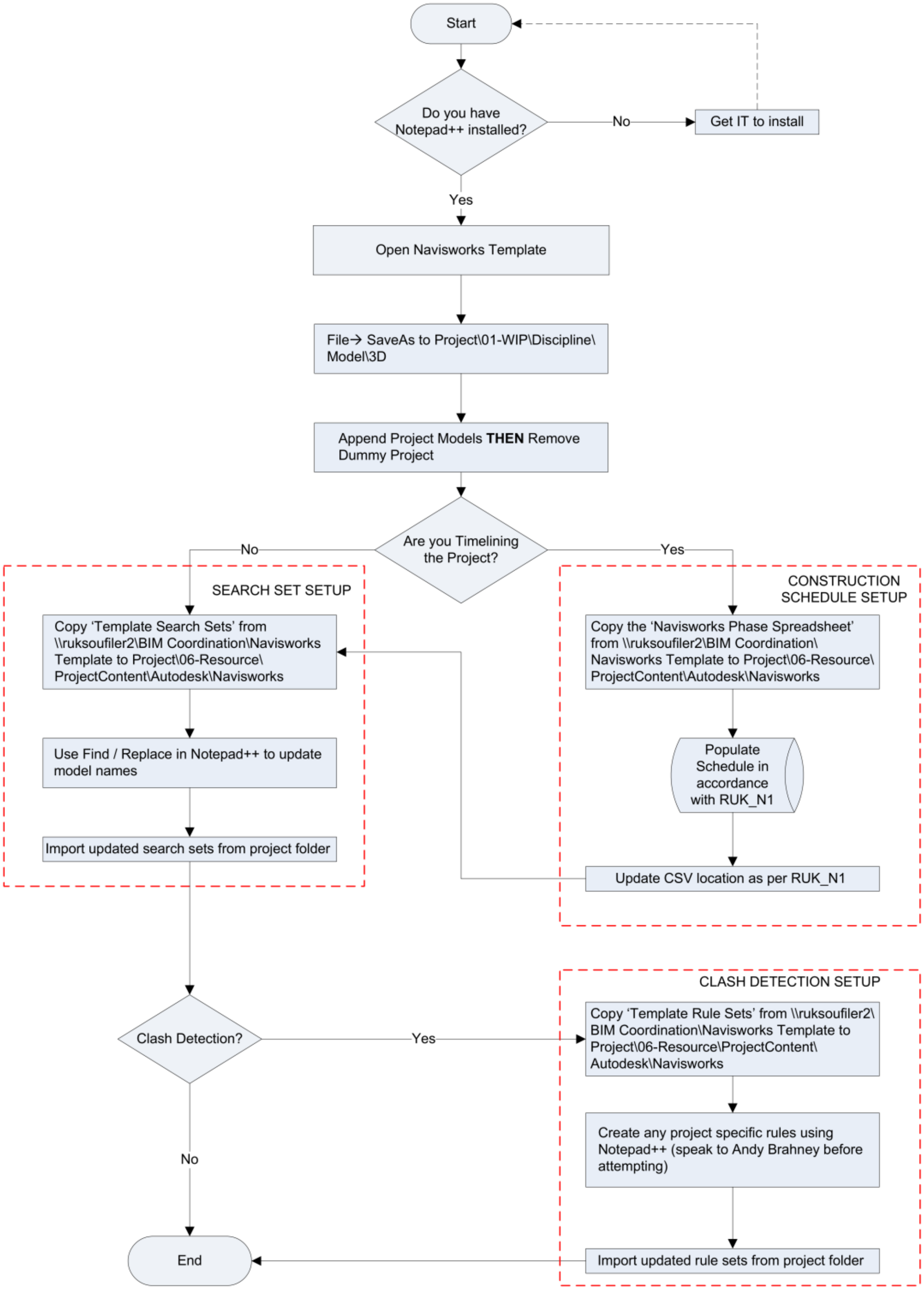
Tip 4 – Company Logo:

To replace the Autodesk logo in the top left of the clash report, simply open the HTML file in Excel, right click and select Change picture, then select an image of your company logo.

Develop a Standardised Workflow for Clash Detection

34. The first step in this process is to modify the files we created in the above processes. This is summarised in the workflow diagram below:

RUK_N2 – Navisworks Template Job Specific Setup


Start
Do you have Notepad++ installed?
No
Get IT to install
Yes
Open Navisworks Template
File→ SaveAs to Project\01-WIP\Discipline\Model\3D
Append Project Models **THEN** Remove Dummy Project
Are you Timelining the Project?
No
Yes
SEARCH SET SETUP
Copy 'Template Search Sets' from \\ruksoufiler2\BIM Coordination\Navisworks Template to Project\06-Resource\ProjectContent\Autodesk\Navisworks
Use Find / Replace in Notepad++ to update model names
Import updated search sets from project folder
CONSTRUCTION SCHEDULE SETUP
Copy the 'Navisworks Phase Spreadsheet' from \\ruksoufiler2\BIM Coordination\Navisworks Template to Project\06-Resource\ProjectContent\Autodesk\Navisworks
Populate Schedule in accordance with RUK_N1
Update CSV location as per RUK_N1
Clash Detection?
Yes
No
CLASH DETECTION SETUP
Copy 'Template Rule Sets' from \\ruksoufiler2\BIM Coordination\Navisworks Template to Project\06-Resource\ProjectContent\Autodesk\Navisworks
Create any project specific rules using Notepad++ (speak to Andy Brahney before attempting)
Import updated rule sets from project folder
End

RUK_N2 – Navisworks Template Job Specific Setup	07/04/14	Approved By: GS	Status: Draft	Page 1 of 1

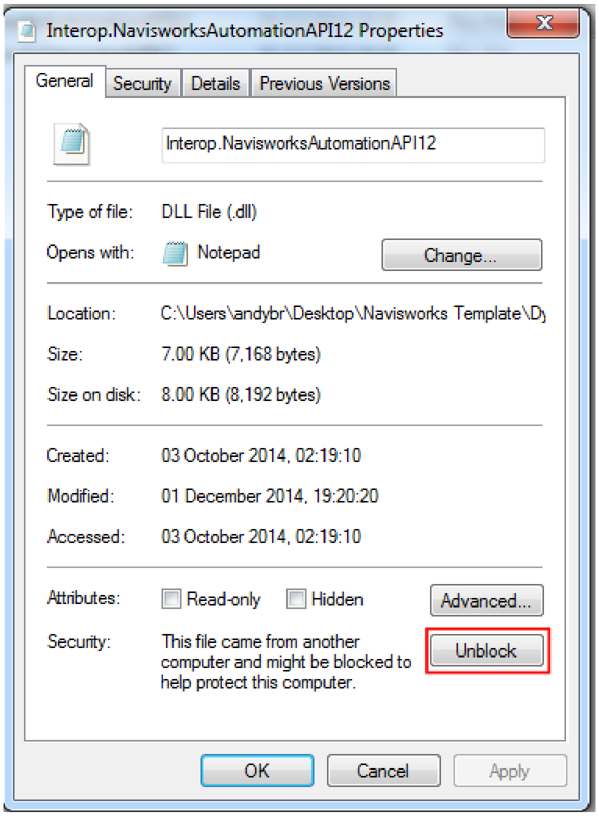
35. Save the file as an NWF in a project specific location. Again, this should not be moved once setup for the following steps to work.
36. Run clash detection!

Now that we have the search sets and clash tests looking at a single, project specific NWF file, we can begin to look at how we go about the clash detection process in an efficient manner. To do this, we shall be using Dynaworks which allows Dynamo to control Navisworks. This process has a number of advantages such as being able to run Navisworks without showing the window and automatically carrying out the clash detection process for us. The program cannot spot false positives, but will carry out the tests for a user to then go in and review. This could be scheduled to be carried out overnight so that the data is available first thing in the morning.

The process of how to install Dynaworks can be found at the beginning of this document.

Before we fire up the program, we need to import the library files to enable Dynamo to utilise Navisworks. These can be downloaded from the Dynamo packages website as outlined at the beginning of this document. To import the modules:

37. Go to the Dynamo Library Modules and copy ALL files into ‘C:\Users\(Username)\AppData\Roaming\Dynamo\0.7\packages\DynaWorks15’
38. Right click the DLL files and ensure that you hit ‘Unblock’ if the option is available:



Interop.NavisworksAutomationAPI12

Type of file: DLL File (.dll)

Opens with: Notepad

Location: C:\Users\andybr\Desktop\Navisworks Template\Dy

Size: 7.00 KB (7,168 bytes)

Size on disk: 8.00 KB (8,192 bytes)

Created: 03 October 2014, 02:19:10

Modified: 01 December 2014, 19:20:20

Accessed: 03 October 2014, 02:19:10

Attributes: □ Read-only □ Hidden

Security: This file came from another computer and might be blocked to help protect this computer.

Fig. 4.1 – DLL Properties Dialog

39. Launch Dynaworks from the Start Menu, All Programs, Dynamo.

You shall then be given the following splash screen:

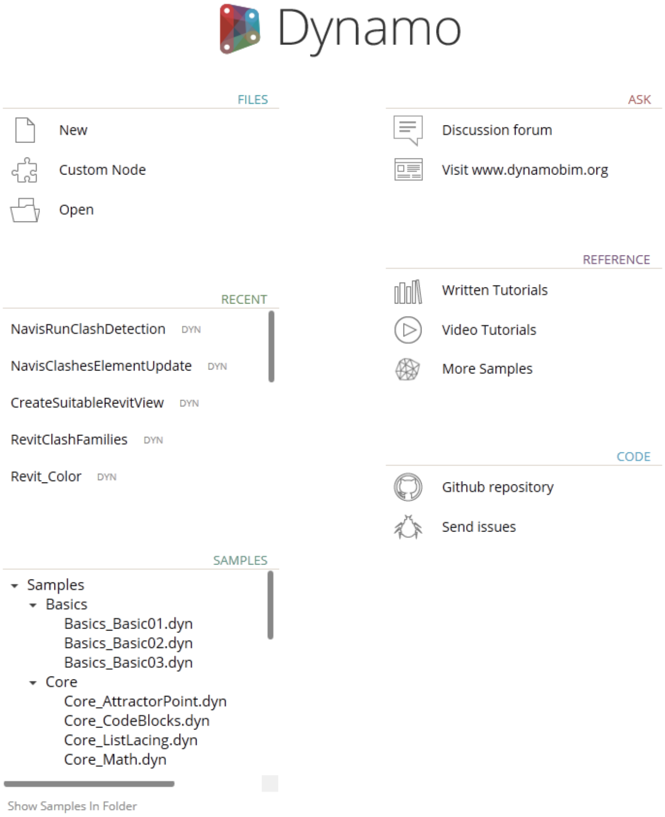


Fig. 4.2 – Dynamo Splash Screen

We now need to create the Dymano workflow to carry out the clash detection. Luckily for us, Adam Sheather of the ‘Stuff & BIMs’ blog who worked on developing Dynaworks, has done some of the heavy lifting for us and provided Dynamo definitions which can be downloaded here.

40. Select Open and navigate to the Dynamo Definitions folder.
41. Open the ‘NavisRunClashDetection’ file.

The file structure looks like this:

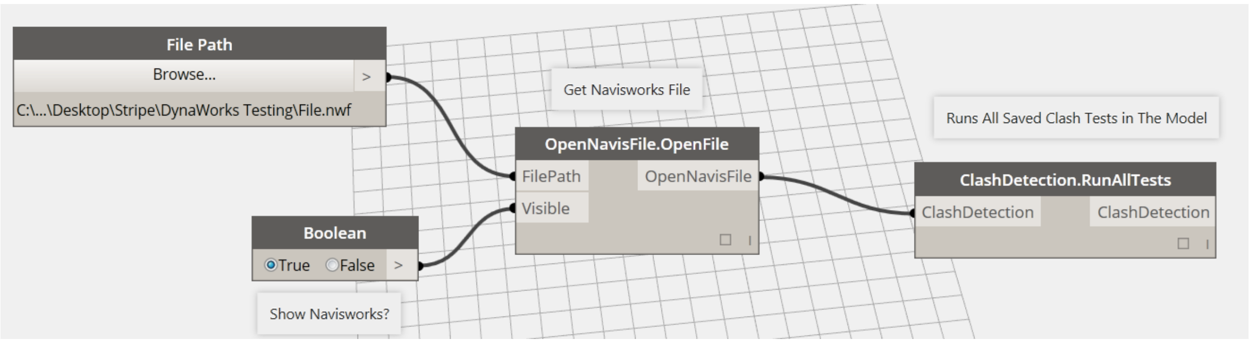


Fig. 4.3 – NavisRunClashDetection File Structure

The nodes each have a description attached to give further clarity on the function of each. Unfortunately, there is a missing node in this file, which we will need to add before we carry out the detection.

42. Insert the GetClashDetection node to the workspace. In the library menu on the left hand side, expand the Dynaworks15 tab, and left click ‘GetClashDetection’ as shown below:

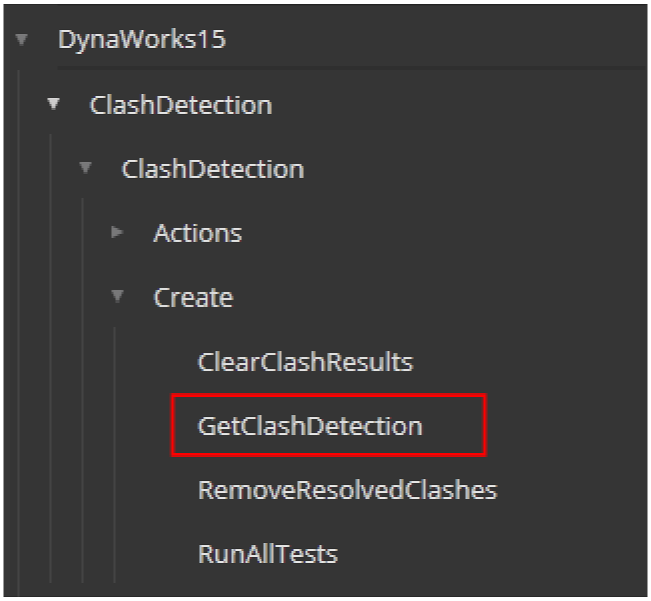


Fig. 4.4 – GetClashDetection Node

43. Drag the node so it is in between ‘OpenNavisFile.Open’ and ‘ClashDetection.RunAllTests’.

44. Connect the nodes by clicking the buttons highlighted below, starting from left to right:

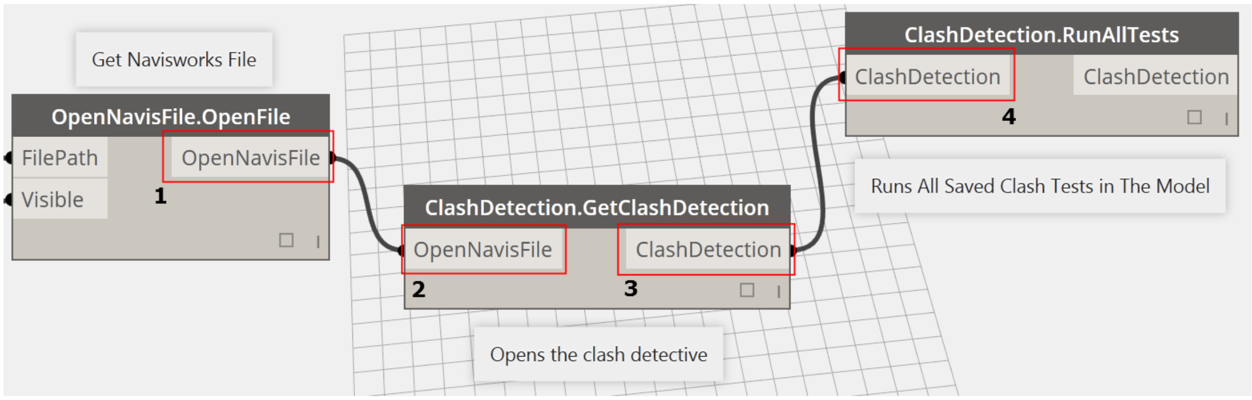


Fig. 4.5 – Connecting New Node

45. Update the File Path to reflect the location of your DWF file by pressing the 'Browse…' button on the File Path node.
46. Save the file.
47. Close Dynaworks.

Now that we have this definition set up, we can use this to clash detect multiple projects at once. To do this:

48. Open Dynamo.
49. Select all of the nodes in the file and press Ctrl+C to copy them to the clipboard.
50. Press Ctrl+V to paste the elements in the same workspace, and move them so they don't overlap.
51. Update the File Name in the copy to another model you wish to clash.

Be aware that this will launch an instance of Navisworks for each model you are clashing and carries them out concurrently which will be very resource intensive. Arranging Dynamo to clash the models sequentially would be a better approach, however this is beyond my Dynamo skills, so if you have a better way to do this, please do share!

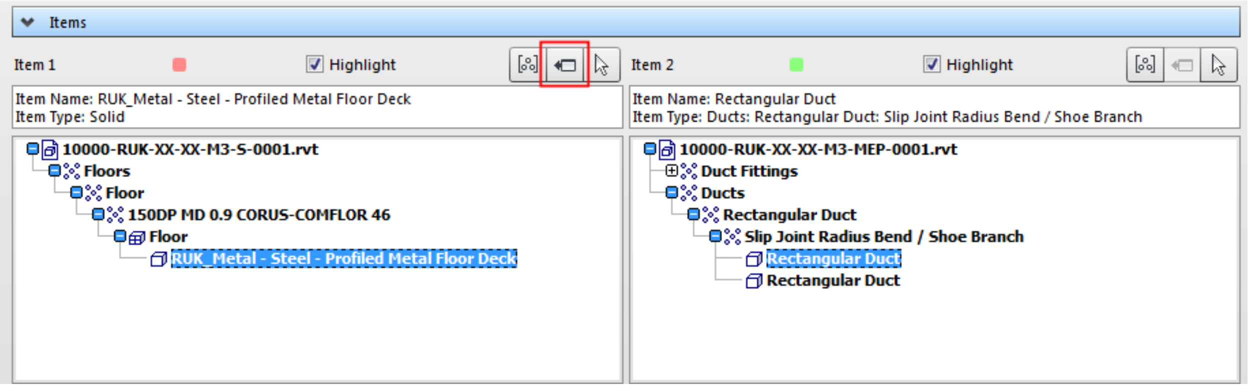
So what do we do with the output from the clash detection process? Revit has an add-in called Switchback which, as the name suggests, allows you to select a clash in Navisworks and then have the program take you to it in the Revit model. One workflow for this is:

52. Launch the Structural Revit model you will be editing.
53. Navigate to the 'Add-ins' tab on the Ribbon.
54. Under 'External Tools' select 'Navisworks Switchback 2015'.

When you press the button, it will appear as if nothing has changed, but this action allows Navisworks to interact with Revit.

55. Launch the clash NWF in Navisworks.
56. Open the Clash Detective and go to the Results tab.

57. Expand the Items menu, as shown in Fig.3.1
58. Select the switchback button for the element as highlighted below:



Item 1: ☑ Highlight

Item Name: RUK_Metal - Steel - Profiled Metal Floor Deck
Item Type: Solid

```
10000-RUK-XX-XX-M3-S-0001.rvt
  Floors
    Floor
      150DP MD 0.9 CORUS-COMFLOR 46
        Floor
          RUK_Metal - Steel - Profiled Metal Floor Deck
```

Item 2: ☑ Highlight

Item Name: Rectangular Duct
Item Type: Ducts: Rectangular Duct: Slip Joint Radius Bend / Shoe Branch

```
10000-RUK-XX-XX-M3-MEP-0001.rvt
  Duct Fittings
  Ducts
    Rectangular Duct
      Slip Joint Radius Bend / Shoe Branch
        Rectangular Duct
        Rectangular Duct
```

Fig. 4.6 – Switchback Button in Items Menu

59. You will be taken to revit with a zoomed in view of the element selected.

From here, you can then work through the model to correct the clashes.

60. After making the changes, save the Revit file.
61. Go back to Navisworks and press the Refresh button on the home tab.

If you are carrying out clash detection on a regular basis, it may be worthwhile to utilise the Batch Utility in Navisworks which can be scheduled to append multiple files to an NWF automatically. This topic is covered on the Navisworks help website:

http://help.autodesk.com/view/NAV/2015/ENU/?guid=GUID-974E2165-7403-4025-B0D0-F7EBC46AC592

Future Development

There is a lot of potential for future development of your Navisworks template, for instance, at Ramboll we have incorporated the timeliner functionality which enables us to produce construction sequences more quickly.

With respect to the clash detection, you can use Dynaworks to insert an object at each clash point. This is also covered the Stuff & BIMs blog; however, in my experience, the clash points are placed in the model with an elevation from the Project Origin. This means that the points come in in the wrong elevation. Hopefully, this is a problem that will be addressed moving forward.

Finally, as you may have noticed in my lab, I am not a big fan of the clash reports: I find them static and not very user friendly. I would like to see some kind of interactive dashboard, which some companies have produced, that shows very quickly the number of clashes in the model over time, cost savings made (may be too complex), and who has the most clashes. Data mining is already being carried out using Dynamo, so this may be done using a more advanced version of Dynaworks.

Summary

In this lab, we have looked at a few of the ways in which you can save time and money in carrying out clash detection in Navis. The aim of the lab was not to be exhaustive, but to display processes which can very easily be adapted to a whole number of other uses both in Navisworks and other packages.

I hope you have found the lab useful, and that you have lots of ideas flying around your head about where you see this going! If you have any questions or ideas, please do get in touch with me and we can geek out together!

Thanks, and enjoy the rest of AU!