



## Customizing Visual Intercept Studio for TestDirector

Visual Intercept Studio for TestDirector uses an XML template to allow a user to customize the format and content of the information shared between Visual Intercept and Mercury Interactive's TestDirector. This month's TechTip describes how to customize your Visual Intercept Studio for TestDirector integration to fit your needs.

The type and format of information shared between Visual Intercept and Mercury Interactive TestDirector is defined in a template file named viETTIncident.xml which is located in the primary Visual Intercept installation directory. The default location of this directory is Program Files\Visual Intercept Enterprise\Template. To customize the integration between the two applications simply edit this file based on the following concepts.

### How does the integration work?

The integration between Visual Intercept and Mercury Interactive TestDirector is structured so that an incident can be registered from TestDirector by simply clicking the Add Defect button in the TestDirector IDE. This will open a Visual Intercept incident form that is populated with the available TestDirector information. After the incident has been submitted to Visual Intercept, a defect is submitted back into TestDirector based on information returned from Visual Intercept. This process uses the viETTIncident.xml file to determine which field values are to be moved between the two systems and how the values should be mapped between the two schemas.

### What is the format of the viETTIncident.xml file?

The viETTIncident.xml file follows a standard XML format with sections called nodes that define the fields and their mappings that are to be used for three distinct situations. The nodes defined by the **<vilIncident>** tag are used for moving information from TestDirector into Visual Intercept. The node defined by the **<tdDefect>** tag is used for moving information from Visual Intercept back to TestDirector.

There are two **<vilIncident>** tags because there are two possible contexts under which a user may submit a defect from TestDirector to Visual Intercept. If the user submits a defect while looking at a specific Test Run that has failed, the vilIncident node with **Name=tdTestRun** will be used otherwise the vilIncident node with the Name attribute of **Name=tdBug** will be used. This allows the user to format a more complete incident for Visual Intercept in the situation where extended Test Run information is available. In either case, Visual Intercept returns information defined by the **<tdDefect>** node back to TestDirector.

Within a particular node are tags called elements that define fields that should be populated with information. In these elements, a user can place literal text as well as substitution variables to define the type and format of data to be included in these fields. The substitution variables are defined by the \$ symbol followed by the actual column name where the values will come from. These substitution variables will be replaced at runtime with the actual values.

### Configuring <vilIncident> nodes

The **<vilIncident>** nodes should be configured with the intent of including TestDirector information into the Visual Intercept incident. The elements are defined by the Visual Intercept database. For example, the information in the **<Description>** element would show up under "Description" in the Visual Intercept form. You could add a **<ProjectName>** element and set the value to be a known Visual Intercept Project name that you want all your TestDirector incidents to be registered under or add a **<Category>** element and set a default Visual Intercept Category to indicate the incident was generated from a TestDirector test run.

### Configuring the <tdDefect> node

The **<tdDefect>** node should be configured with the intent of including Visual Intercept information in the TestDirector defect. The elements are defined by the TestDirector database. For example, the information in the **<BG\_SUMMARY>** element would show up under "Summary" in the TestDirector defect form. You would typically edit this node to reflect default information where the Visual Intercept values were different from the TestDirector values, such as Category, Priority, or AssignID.

### An Example

This example will edit the **<vilIncident>** node so that the active TestDirector Project is used for the subject line for new Visual Intercept incident. The **<vilIncident>** node's **<Subject>** element will include the substitution variable \$TD\_PROJECT so that, at runtime, Visual Intercept Studio for Mercury Interactive TestDirector will replace the text with the actual value.

1. Open the viETTIncident.xml file in a text editor
2. Locate the vilIncident node with the attribute Name=tdBug
3. Locate the **<Subject>** element within that vilIncident node
4. Between the **<Subject>** tags insert the text \$TD\_PROJECT
5. Save the viETTIncident.xml file
6. Select the "Add Defect" button from TestDirector

You should now see the active TestDirector project name displayed in the Visual Intercept incident forms "Subject" line.

This information is also included in the Visual Intercept Studio online help. If you have other questions about issues not included in or beyond the scope of this Tech Tip, please contact Elsinore Technical Support Services at support@elsitech.com or 866.866.0034, option 2.