


<h1 style="color: blue; margin: 0;">SCR (Bug Reports) Guidelines</h1>	
<b>Owner:</b>	Rafael Soultanov
<b>Audience:</b>	SolovatSoft QA Team
<b>Version / Status:</b>	9.0
<b>Location:</b>	San Francisco
<b>Updated:</b>	20/10/2005
<b>Last Printed:</b>	

## Table of Contents

SCR Guidelines. 1  
 Table of Contents. 1

1. Overview.... 1
2. SCR Priorities. 1
3. General Troubleshooting Guidelines. 1
  1. Investigating an SCR... 2
  2. Clearing an SCR... 2
  4. Assignments.. 2

### 1. Overview

This document provides some guidelines for the re architecture project for client / SolovatSoft QA teams and Project management from both companies working with SCRs. This is relevant primarily when a the new re architecture project release has been delivered to QA and is in the process of heavy testing.

### 2. SCR Priorities

During heavy testing phases, all tasks should come from SCRs, unless specific exceptions are made.

When assigned to a developer, every SCR should contain a valid Report Type and Fix Priority. If you receive an SCR with a confusing Fix Priority setting or no priority, contact the appropriate Manager (should be defined) to resolve. Use this chart to organize your work:

Report	Type	Fix Priority
Defect	Critical	Work on it right away; these are usually build issues Or other infrastructural problems
Defect	High	Takes priority over all other work unless specifically

Change Request	High	instructed otherwise Change request are frozen during the testing, after the “UI freeze” date
Change Request	Medium	DefectMedium
DefectLow	Change Request Low	We do it if we can

Definitions of the Fix Priority and Severity fields can be found here:

[http://tnet2/qa/Info/Definition\\_of\\_Severity.htm](http://tnet2/qa/Info/Definition_of_Severity.htm) (should be placed there)

[http://tnet2/qa/Info/Definition\\_of\\_Priority.htm](http://tnet2/qa/Info/Definition_of_Priority.htm) (should be placed there)

### 3. General Troubleshooting Guidelines

With parallel development, we must be careful to follow these guidelines. Adherence to this procedure is important to maintaining stable systems in QA and tracking all the changes that are made.

During the primary development cycle, when requirements and designs are being implemented, developers will work from various documentation such as Use Cases, mockups and other specifications from product development. In some cases, technical design documents that cover the data structures and business objects will be provided. During this time, bug tracking database is not typically used by developers.

After the first build to QA for a specific release, developers will be required to monitor their “In Tray” in bug tracking databases regularly.

#### 1. Investigating an SCR

Be sure to read the description carefully. Pay close attention to the Deployment Platform field – it tells you where the problem occurred, and the Frequency field – it tells you if the problem is always reproducible or intermittent.

Always try to reproduce the problem on the Development system. If it is reproducible there, then fix it and move on. If it is not, then reproduce the problem on the platform where it occurred. You may need help to monitor IIS or App logs to determine why the bug occurs only on that platform.

If all attempts to reproduce the SCR fail, speak directly to the submitter of the SCR. Ask them to reproduce for you. If they cannot, they you can mark the SCR as “not reproducible” (see below).

#### 2. Clearing an SCR

The most common way to clear an SCR is to fix the problem (assuming you could reproduce it). Follow these steps:

From PVCS, open SCR

Add any Notes as necessary.  
Set "Current Resolution" to Bug Verify  
Set "Current Department" to SQA  
Set "Current Owner" to the Submitter. If the Submitter is NOT from SQA dept,  
set "Current Owner" to QA Manager  
Set "Verify in Build" to the next build number. To determine the next build number,  
add the number of Week Days to the last build number  
Click OK  
Hit Shift-F5 to remove the SCR from your In Tray.

If you could not reproduce the SCR on the platform/environment it was found then  
follow the same steps except:

- Set "Current Resolution" to "Not Reproducible"  
Leave "Verify in Build" blank

NOTE: Its not acceptable to mark an SCR as "Not Reproducible" just because it works  
OK on DEVSRVxx. If the defect is caused by a build problem it is still our responsibility  
to determine where the specific problem lies.

If the SCR describes a problem that was known during development but it was decided  
that it would not be resolved, then follow the same steps except:

- Set "Current Resolution" to "Working As Designed"  
Leave "Verify in Build" blank