SFEE 3.4 File Publisher Frequently Asked Questions

1. **Are there additional File Publisher features available in SourceForge Enterprise Edition 3.4?**

   No. The SourceForge File Publisher functionality is the same in SFEE 3.4 as in SFEE 3.3.

2. **What is "Maturity"?**

   A File Release can be associated with a Maturity level. This attribute is intended to indicate the state of the Release.

   For example, a Release can be designated "Alpha", "Beta", or "Stable".

3. **Are Maturity levels fixed?**

   No, a File Publisher Admin (File Publisher administrator) can create custom Maturity levels.

4. **Do the same Maturity levels apply across a SourceForge installation?**

   No, Maturity levels are project-specific. Each project can have a different set of Maturity levels.

5. **Are Maturity levels mandatory?**

   No, Maturity levels are optional.

6. **After the File Publisher administrator creates a set of Maturity levels for a project, should every subsequent File Release be assigned a Maturity level?**

   No, File Releases can be uploaded without setting a Maturity level, even when the project has Maturity levels defined.

7. **What are "Dependencies"?**

   If the use or operation of a File Release depends on other File Releases, the File Publisher administrator can set the dependencies when creating a File Release. This indicates to the user that if a particular File Release is to be downloaded and used, the other File Releases in the dependency list may also be required.

8. **Are dependencies between File Releases limited to within a project?**

   No, File Releases can have dependencies on any other File Releases in any SourceForge project.

9. **What are SCM Tags?**

   Most SCMs provide tagging ability for checked in files. Tagging all the files of a Release with the Release name is a common practice. When using SourceForge to distribute File Releases, the File Publisher administrator can indicate which tags are associated in the SCMs with a given File Release.

10. **Are SCM tag associations mandatory?**

    No, the tag associations are optional.

11. **What is the Access Log?**

    SourceForge File Publisher keeps a comprehensive log of file access. "Username", "Date of Download", "Time of Download", "User Agent", "IP Address", and "Number of Downloads" are tracked for each file. The access log contains this information, collated for each File Release.

12. **Can anybody view the log?**

    No, only a File Publisher administrator can view the access log. Other users cannot view the access log.

13. **Can the access log be turned off?**

    No, all downloads are logged. Logging cannot be turned off.

14. **What are Artifacts discovered and Artifacts addressed?**

    A Tracker artifact can be associated with a File Release. The association can be made in two fields: "Discovered in" and "Addressed in". **For example**, if a bug was discovered in Release 3.1.2 and fixed in Release 3.2, this can be indicated in the respective fields when creating or editing the Tracker artifact. From the File Release interface, users can view the associations.

15. **Can the associations to a Tracker artifact be made from within the File Publisher interface?**

    No, associations are created and maintained from within the Tracker system. The File Publisher interface provides the user with a variety of ways to view the associations.

16. **How can I integrate External Build systems with the File Publisher?**

    SourceForge provides an XML RPC hook that build systems can use to automate the population of a File Release. Most build systems provides a scripting facility. Using this scripting facility, an XML RPC call must be made to transfer a single file to the File Publisher.

17. **What is the XML RPC call I need to use to transfer a file to the SourceForge File Publisher ?**

    The XML RPC hook provided by the File Publisher is a single procedure call. The files to be uploaded into the File Publisher are passed as parameters. The method signature is as follows:

    string X frs.fileUpload(struct A,string B,string C,string D,string E,string F, string G, string H, string I)

    X - Returns a string Success Message or an Error message depending on success or failure. The success message will be "File Release Updated".

    struct A – This is an associative array with the file name as the key, the base64 encoded file as the value.

    string B - user name

    string C – password

    string D - project short name

    string E - package name

    string F - file release name

    string G - maturity

    string H – changelog

    string I – release notes

    This will help the build engineer automate the uploading process using an appropriate loop.

18. **What if the Release/Package does not exist in the File Publisher when I transfer a file?**

    If the Release/Package does not exist, a new Release/Package will be created. If the project does not exist, an error message is returned.

19. **Can I skip the username & password, transfer the file to the server, and move it at a later time?**

No, a valid username and the password must be entered before transfering the file. This is to ensure that unauthorized users do not clutter up server space with irrelevant content.

20. **Can the call be made over http?**

    No, the call must be made securely over https.

21. **What is the maximum file size I can upload to the File Publisher?**

    The default maximum file size for upload to the File Publisher is 100MB. If you attempt to upload a file larger than 100MB, the upload will fail. No error message will be provided and you will be returned to the File Publisher Administration page.

    If you would like to increase the maximum file size for upload to the File Publisher, please contact VA Software Product Support for assistance.