|  |  |  |
| --- | --- | --- |
| **Tool** | **Summary** | **Limitations** |
| **Softchange**  **[Ger04]** | Extracts historical data from CVS repositories and defect trackers and joins both sources of information. It can also do static analysis of C++ and Java source code. | Only capable of extracting data from CVS. Source code analysis is syntactic in nature only. No support for structural metrics mining. |
| **Hipikat**  **[CM03]** | Similar to Softchange in that it is designed to join multiple sources of open source project data. Supports CVS, Bugzilla, Newsgroups, and mailing list archives. | Only capable of extracting data from CVS. No static source code analysis. |
| **Dynamine**  **[LZ05]** | This tool correlates repository revision histories with snapshots of source code to identify common code change patterns. | Language and VCS independent but fairly limited in that it is designed with specific use cases in mind – namely to identify common commit patterns across individual repositories. |
| **Kenyon**  **[BWJKG05]** | Kenyon provides a resuable framework to extract data from any number of VCS systems and store it to disk in a customisable file format. | No static source code analysis. No out-of-the-box support for database persistence. |
| **CVSAnaly**  **[Rob]** | Extracts information from source code revision history logs and stores them in a database. Supports multiple VCS systems. Active research community. | No out-of-the-box support for static Java source code analysis. |