LexEVS Software Release Versioning

Document Information

Author: Craig Stancl, Scott Bauer, Cory Endle
Email: Stancl.craig@mayo.edu, bauer.scott@mayo.edu, endle.cory@mayo.edu
Team: LexEVS
Contract:
Client: NCI CBIIT
National Institutes of Heath
US Department of Health and Human Services

Contents of this Page

- Version Identification
  - References

The **purpose of this document** is to document the software identifiers for the National Cancer Institute Center for Biomedical Informatics and Information Technology (NCI CBIIT) **LexEVS**.

Version Identification

LexEVS and associated projects will follow the following version model:

```
major.minor.patch[.maintenance[.build]]
```

Given a version number MAJOR.MINOR.PATCH.MAINTENANCE.BUILD, increment the:

1. **MAJOR** version when you make incompatible API changes, model changes or updates to external (i.e. Java, OS) technology (Tier Deployed)
2. **MINOR** version when you add functionality in a backwards-compatible manner or updates to internal (i.e. Lucene, DB) technology (Tier Deployed)
3. **PATCH** version when you make backwards-compatible bug fixes. (Tier Deployed)
4. **MAINTENANCE** version when you make changes to loaders, admin scripts, etc. (Not tier deployed but public)
5. **BUILD** version intended for testing and internal use only. (Not tier deployed)

References

http://semver.org/
http://inedo.com/blog/how-to-create-version-numbers-that-are-actually-semantic
https://en.wikipedia.org/wiki/Software_versioning