Selena roadmap

Planned for final release

Project Configuration

- Smarter project setup
  - Detection of existing libraries
  - Detection of JavaEE aspects
  - Selectable default project layout (Maven etc)
- Rework of Project Settings UI
- Directory-based project settings storage
  - No project reload for most settings changes

Deployment configuration

- Configurable packaging of project output (generalization of Web module deployment options)

Dependency management

- Dependency structure matrix
- Suggestions on what dependencies to get rid off with maximum effect

TestNG integration

- Adaptation of existing plugin

Maven integration

- Project structure synchronization
- Execution

Version controls

- Generic support for offline VCS operations
- Incoming changes toolwindow (up-to-date list of changes recently committed by other users)
- ClearCase UCM support
- Directory Diff

Refactorings

- Extract constructor (from another constructor)
- Move class into other class
- Inline (inner) class as anonymous
- Replace method with method object (aka builder pattern)
- Extract method to look for method call duplicates across the project
- Extract class
  - Extract bunch of fields into separate class with bean access autogenerated
  - Extract independent part of functionality with delegation from original class
- Change type
  - Select bunch of type occurrences and replace them all at once building transitive closure based on type flow.
  - Replace array with generic collection (list). Should be aware of toArray() pattern and avoid collection->array->collection conversion

Code Generation

- Generate visitor support for class hierarchy

Peformance improvements

- Faster startup and project opening
  - Local History rewrite
  - VFS rewrite

Update management

- Enterprise plugin repository
- IDEA Auto update
- Silent install
- Plugin auto update, plugin configuration export/import (for large teams)
- Plugin enable/disable
- "Safe startup mode"
- Search in Plugin Manager dialog

Java EE

- Migration of Web and Java EE modules to facets
- Database/mapping structure diff for JPA & Hibernate
- Spring patterns
- Spring AOP
- Making J2EE an optional plugin
- Web module configuration streamlining/making it more flexible

JavaScript

- Refactorings: extract/inline method
- ECMAScript level 4 support

Misc

- External annotations
  - Annotate library classes (e.g. JDK) with Nullable/NotNull/NonNls (for use by IDEA inspections, no code instrumentation intended).
- Global data flow navigation

Implemented in Milestone 1

Version controls

- ClearCase integration
- Create/Apply Patch
- Shelve Changes (temporarily remove from code and save as patch, then later restore)
- Store per-project mapping of VCS roots to project directories instead of module-based VCS config
- Autodetect VCS configuration for project

Refactorings

- DnD for move/copy in Project View
- Create inner class from usage

Find Usages

- List of recent "Find" results
- Background mode
- Display of code context for selected usage

Find in Path

- Support for scopes
- Find in zipped sources
- Find in library sources

Debugger

- Frame/Watches simultaneous view
- Smart step
- Show function return value (JDK 1.6)
- Automatic watches view

Eclipse Integration

- Import from Eclipse projects
- Direct reading/writing of Eclipse project files mapped to IDEA modules.

Peformance improvements

- JSP editing
- More operations performed in background (Find Usages, VCS update etc.)

JavaME

- Ant script generation
- Resource bundles?

TeamCity Integration

- Download full inspection report for browsing in IDEA
- Apply quickfixes from server
- Instant highlight of duplicates in editor

Java EE

- Spring support
- Hibernate support
- Facets (sub-module project structure) for Spring, JSF, Hibernate, GWT

JavaScript

- JavaScript documentation support