Introducing
FACTORY SCHEMES™
Adaptable software factory Patterns

FACTORY SCHEMES™ 3 Standard Edition
Community & Enterprise
Key Benefits and Features

GECKO Software
http://consulting.byGecko.com
Email: Info@gecko.fr
Phone: (+33) 04 42 26 06 08

Index

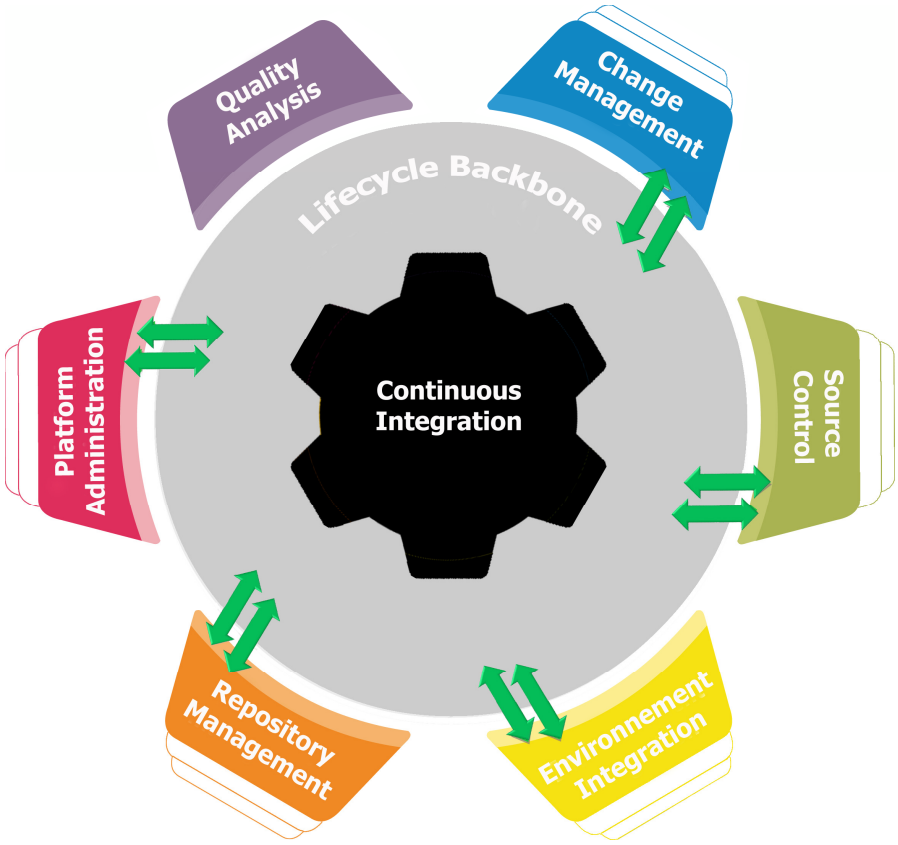
FACTORY SCHEMES™ Key Benefits

Technologies as well as applications development "standards" are increasingly numerous and are more and more frequently subject to changes. If taking into account the changes in **applications construction** is performed manually, the cost increase becomes quickly exponential. This is true even if the source code repositories, development platforms, runtime environments or integration requirements remain the same. It therefore becomes necessary to **industrialize** the manufacturing process of applications. It is also a matter of **"Software Factory".**

FACTORY SCHEMES™ literally provides "Software Factory" plans taking into account the multiple dimensions that are actors, processes and tools involved in manufacturing applications.

These "**FACTORY SCHEMES™**" are customizable given the sensitivities and habits of the company.

FACTORY SCHEMES™ takes advantage of benefits resulting from implementation experimentations, similar in Key Accounts environments (Banks/Insurance Companies/Services).



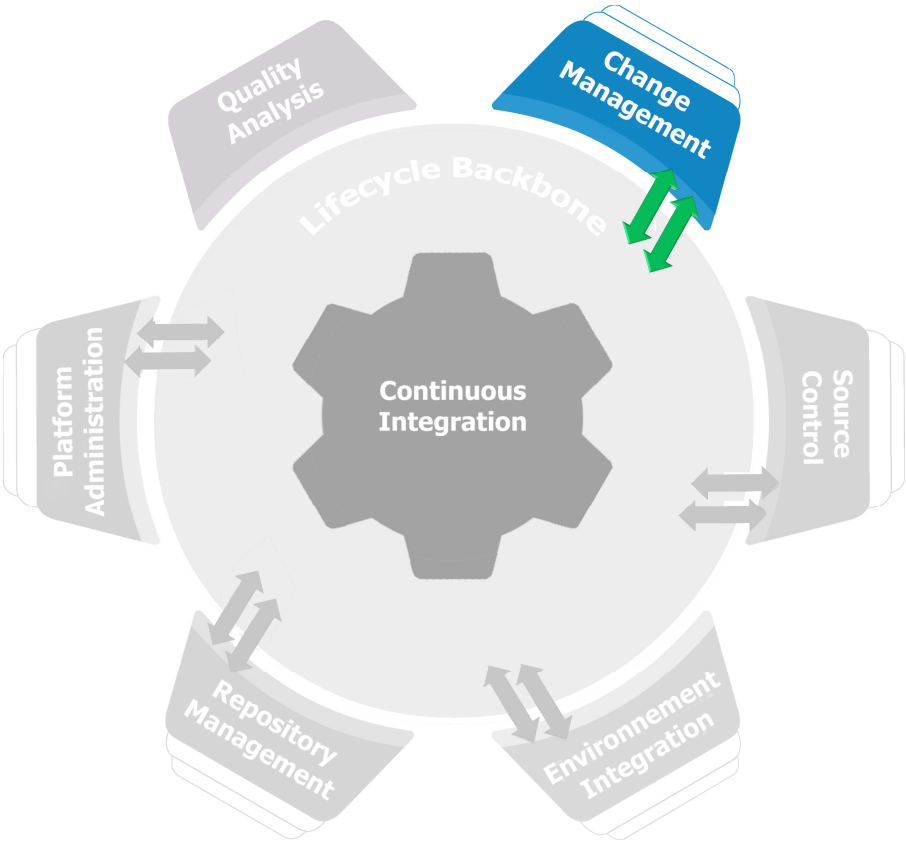
FACTORY SCHEMES™ Key Features

FACTORY SCHEMES™ aggregates industry's best practices coming from concrete continuous integration platforms bringing together all concepts of "Software Factory".

Take a quick tour of **FACTORY SCHEMES™** key features, listed below (soon to be available in Open Source version, already available in Open Source version, and already available in enterprise version), starting with the most global subjects to get to the specific matters of source code's inner quality, of parallel construction method, of application's dependencies maintaining mechanism, etc… This will allow you to analyze our solution by considering your own challenges.

Change Management

Global subject

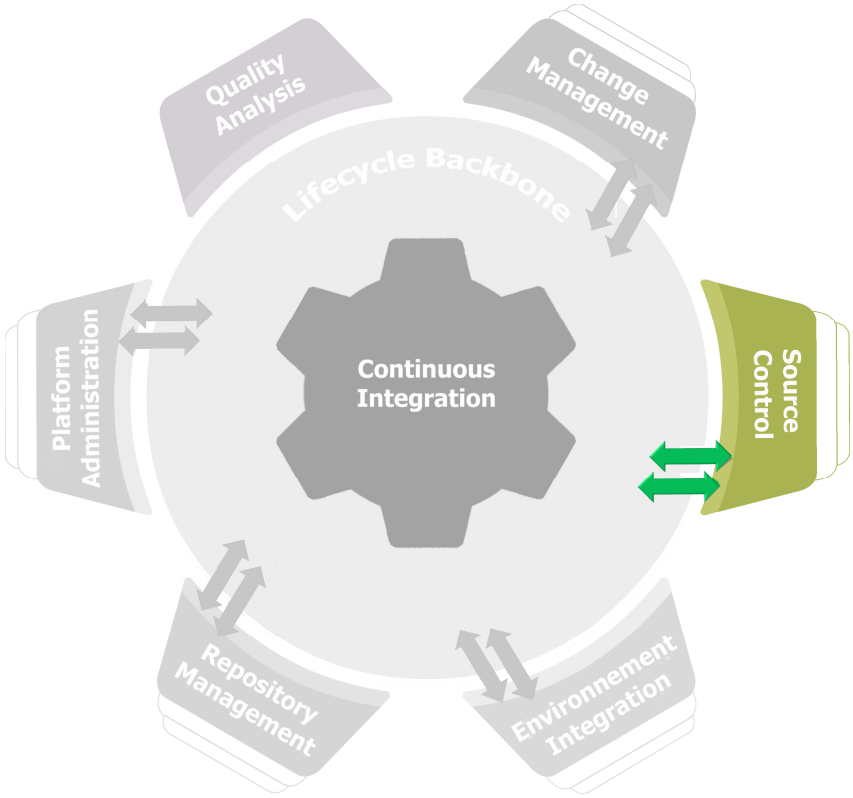


Supported features

Standard fulfillment	- Generation of documentation of a project supported by default Maven reactor and the tracking system (JIRA, Trac, GitHub…): o Issue Management o Release Announcement o Release Notes o Changes Reports…	
Change management	- Normalization of the development process around predefined types of changes to improve visibility across teams and projects : o changes o issues o improvements o features o feedbacks o incidents…	
Ticket management	- Modeling of the development process in a ticket tracking workflow to make sure to take right decisions at the right time, can be configured from an existing workflow or completely redefined with: o custom Fields o email notifications o access controls o permissions...	
Resource control	- Definition of a roadmap organizing each milestone in order to show the path to follow - Resource planning (teams) through the same elements of work (work items) previously defined	
Track progression	- Integration of each new event in the development process to collaborate and stay up to date with the activities of teams and produce an overview (timeline) of the project and tracking progress	

Source Control

Global Subject



Supported Features

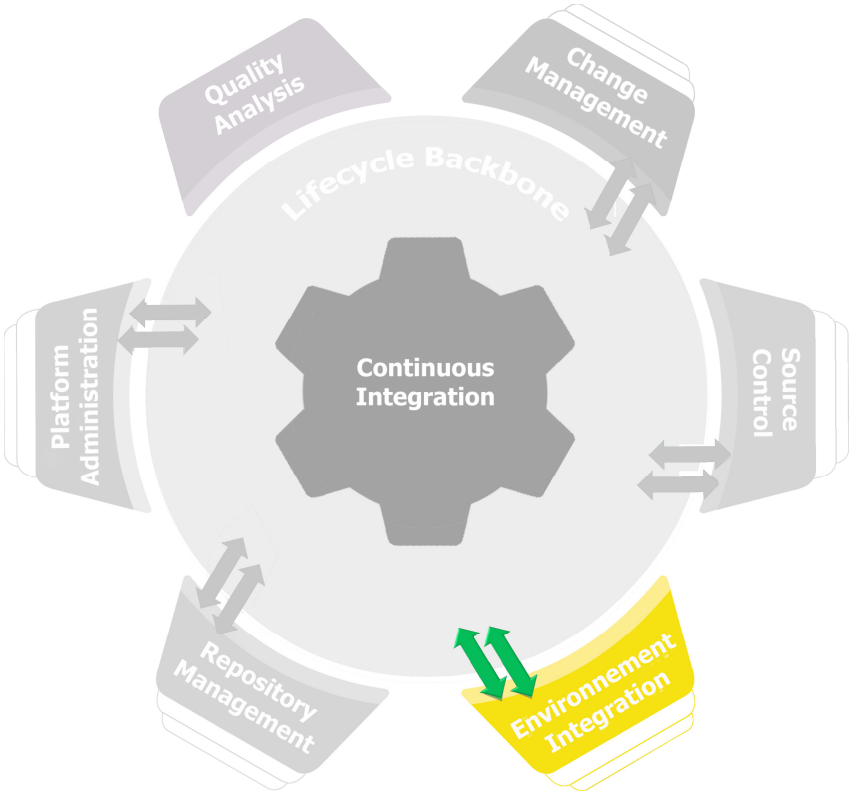
Standard fulfillment	- Maven reactor default version management, as an inner part of the project, and providing a standard mechanism for versioning software configuration changes (also called source control): o SubVersion o CMSynergy o ClearCase o TFS o GIT…	

Seamless commands	- Seamless integration of any specific implementation of common commands with basic functionality to manage each revision resource:
	  o add
	  o changelog
	  o checkin
	  o checkout
	  o update…
	- Seamless integration of any specific implementation of the extended commands providing basic functionality to integrate each revisions:
	  o diff
	  o merge
	  o conflict…	

Features modularity	- Features delegation through specific implementations
	- Specific extension of advanced functions providing wide support for certain uses of project versioning:
	  o release
	  o bootstrap
	  o branch
	  o tag
	  o snapshot…	

Environment Integration

Global Subject



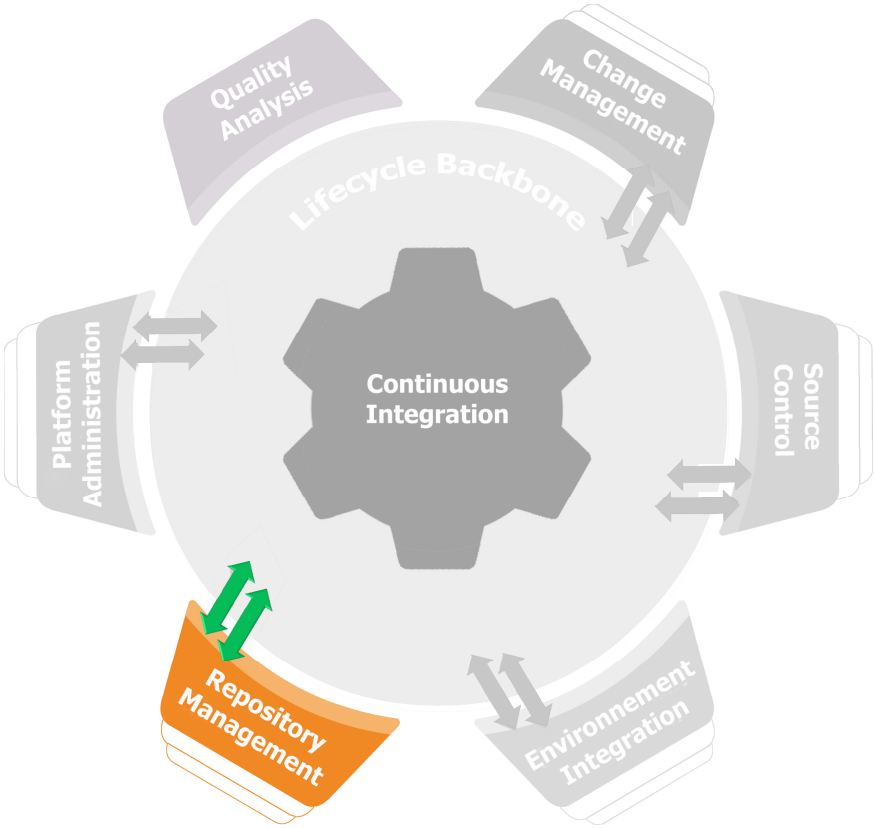
Supported Features

Standard fulfillment	- Transformation of generic meta-models for a specific execution environment:
	  o OMG
	  o MDA
	  o OSGi
	  o JPA…
	- Extraction of generative meta-data for a runtime environment within a specific integration platform:
	  o OS
	  o SGBD
	  o JEE
	  o NET…
	- Generation of descriptive meta-data (files, folders, classpath settings ...) for a specific work environment:
	  o IDE
	  o PDE
	  o RAD
	  o WAS…	

Directing archetypes	- Urbanization of projects through a suitable design of project archetypes, providing a consistent application of inheritance and aggregation for each type of artifact	
Artifacts portability	- Standardization of configuration portability (under certain conditions) of a set of dependencies (potentially different) to adjust the project artifact (e.g., paths, local users or profiles)	
Directing environments	- Generalization of the construction profile concept giving equivalent but different parameters for a set of target environments (e.g. development, test and production)	

Repository Management

Global Subject



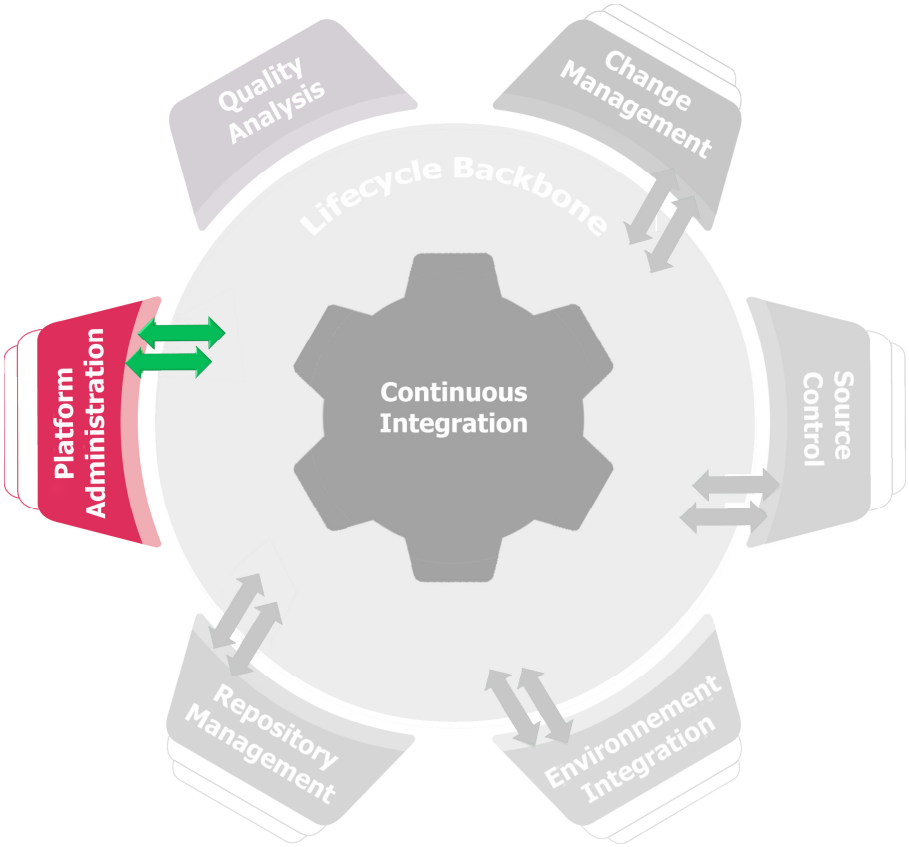
Supported Features

Standard fulfillment	- Organization of default artifacts repositories supported by Maven reactor and repository manager: o Archiva o Artifactory o Nexus…	
Downloads stability	- Definition of remote repository mandated to reflect artifacts downloads to ensure stability within an organization	
Development efficiency	- Concept of local repository, hosted to manage artifacts downloads, in order to promote efficiency and collaboration lifecycle development	

Deployments security	- Definition of deployment repository, staged to ease artifacts downloads to conduct decisions before going into production	
Uniform environments	- Definition of central repository, merged to consolidate artifacts downloads to point to a single storage group	

Platform Administration

Global Subject



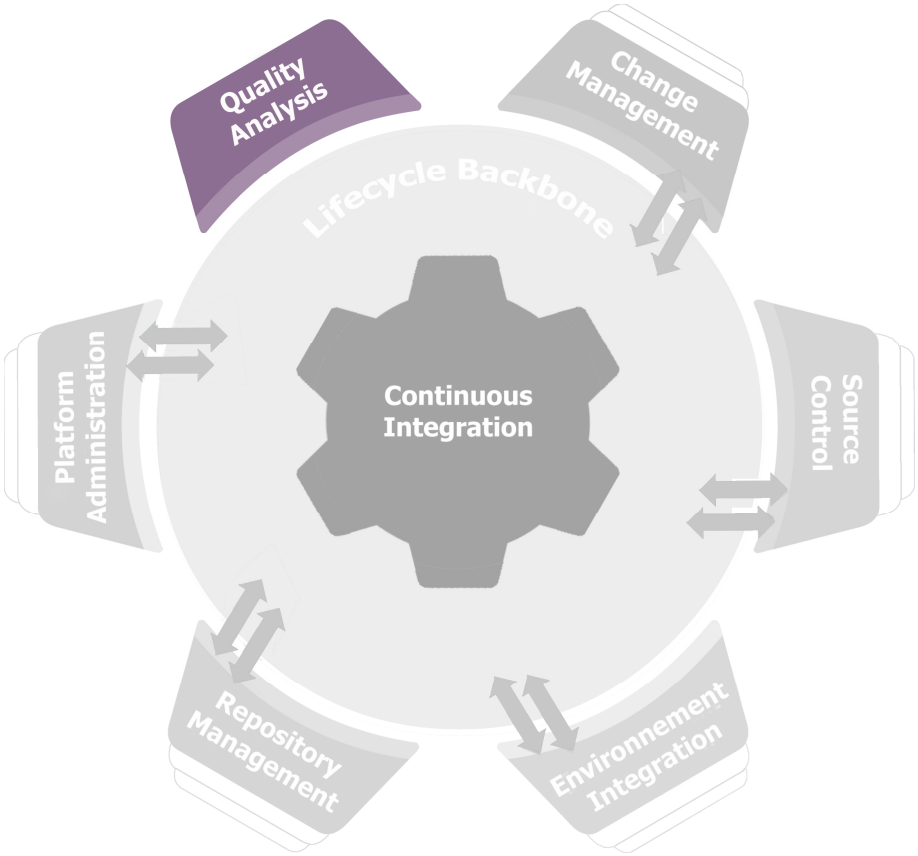
Supported Features

Standard fulfillment	- Administration of platforms supported by default Maven reactor providing a standard interface for installation, configuration, deployment: o Weblogic o Websphere o Tomcat…	
Environments manageability	- Handling environment in a standard way in supporting complex administrative tasks such as: o Restarting a container o Execution of a deployment plan o Plug-in installation …	

Variables portability	- Limitation of portability on a target platform, even within the native solution, through a combination of known variables such as: o operating system o web server o the application container o the server cluster o load balancing o database systems...	
Binaries maintainability	- Distribution of a binary based on a specific target platform and ensuring a certain extent, the maintainability of the binary solution: o centralization o externalization o variabilization o versioning o populating...	

Quality Analysis

Global Subject



Supported Features

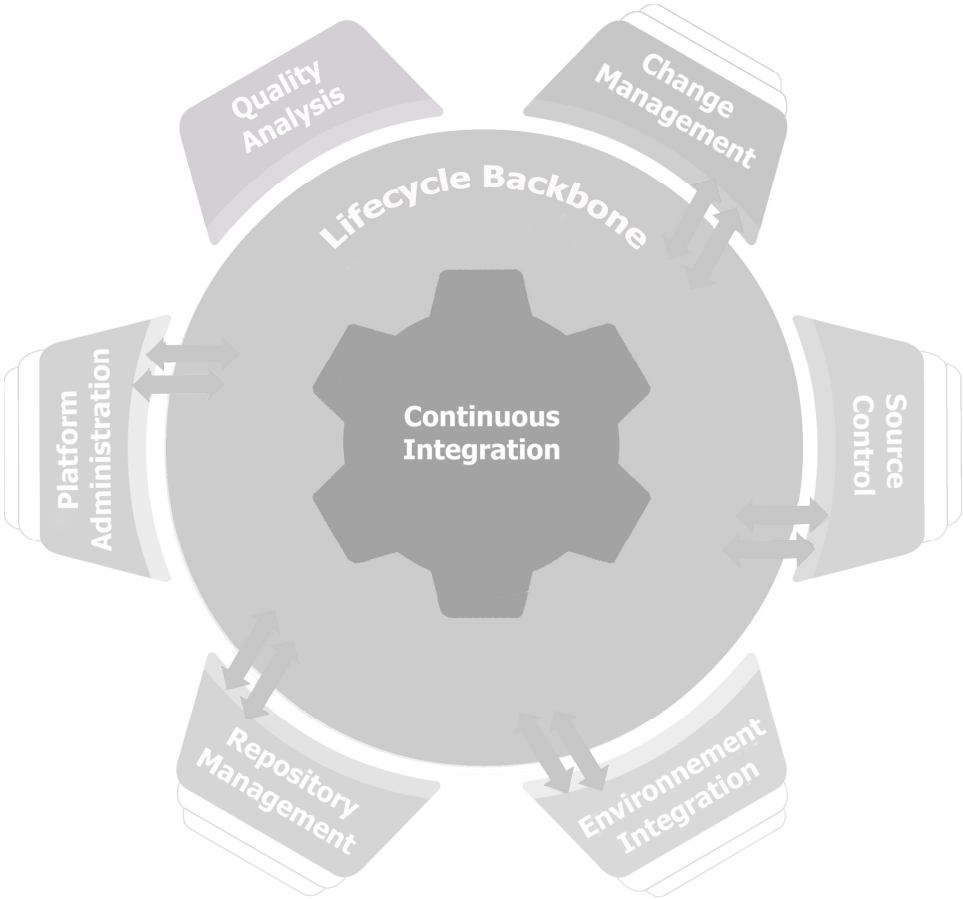
Standard fulfillment	- Aggregation of quality reporting of a project supported by default Maven reactor and analysis system (Hudson, Dashboard, Sonar, Squale, XRadar…): o CheckStyle o PMD o JDepend o FindBugs…	

Conformity of metrics	- Automatic instrumentation of indicators of each design model (method, class, package, module, language, architecture ...) and metric acceptance of a project:
	  o complexity
	  o coupling
	  o cohesion
	  o cycle...
	- Automatic evaluation of technical debt and cost of remediation of non-compliance (clean, refactoring, rewriting ...) application legacy of the company:
	  o portability, maintainability, security, efficiency...
	  o blocker, critical, major, minor ...	

Conformity of requirements	- Introduction of an automatic static code review (duplication, violation, documentation, convention...) to conform application to a predefined set of technical requirements (so-called white box approach) - Introspection of how the application was built (i.e. internal quality) reducing the risk of an application more difficult to maintain over time	
Conformity of specifications	- Automatic execution of a dynamic test coverage (unit, integration, regression, classification ...) to ensure that application meets the functional specifications (called black box approach) - Interaction with application and observation of behavior (i.e. its external quality) reducing the risk of regression over time	

Lifecycle Backbone

Global Subject



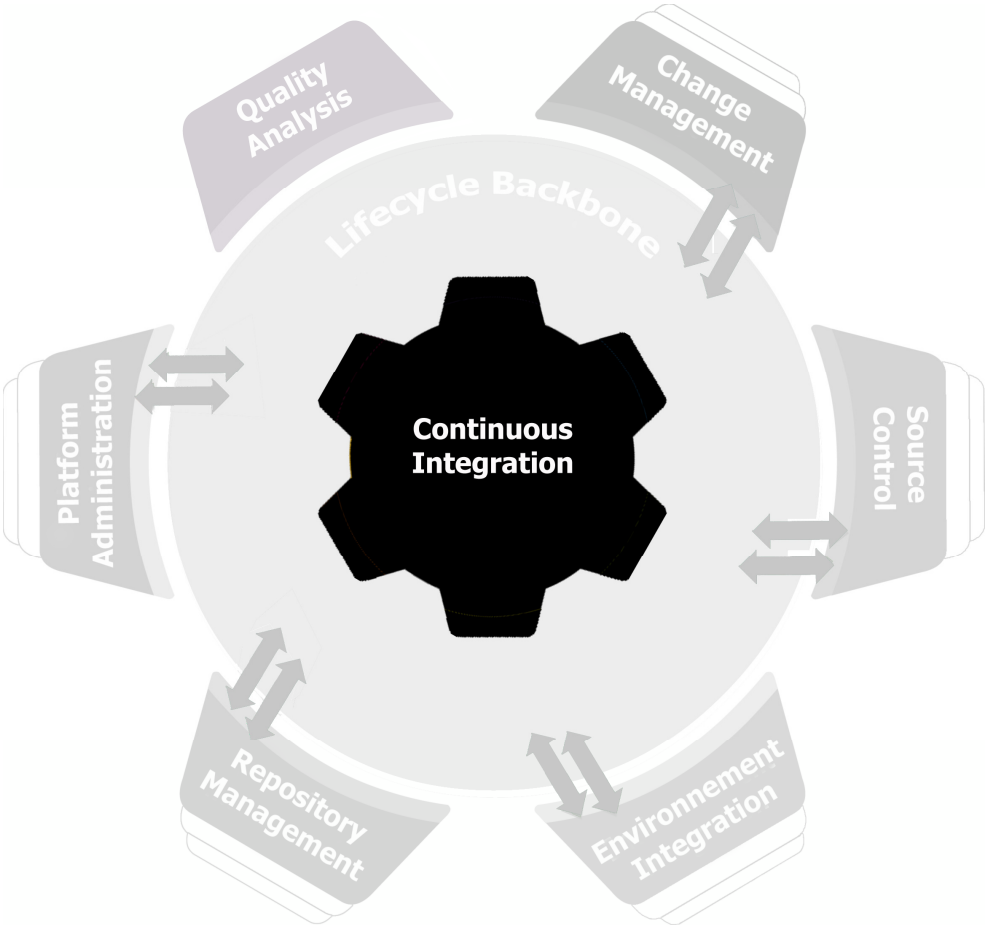
Supported Features

Standard fulfillment	- Modeling of the construction process and distribution of a project (or "artifact") in accordance with all Maven concepts: o Build Lifecycle o Convention Over Configuration o Universal Reuse o POM, MOJO ...	
Segmentation steps (stages)	- Standardization of construction logic around predefined steps of the Maven lifecycle and supported by Maven build kernel (or "reactor"): o clean o compile o package o install o deploy o site…	

Phases independence	- Definition of construction phases sequences, where each phase is responsible for a specific step in the life cycle	
Goals autonomy (goals)	- Factoring responsibilities of life cycle through targets (called "goals") associated with each phase of construction - Isolation of building goals may vary depending on each phase	
Capabilities modularity	- Delegation of life cycle capabilities through integration and / or implementation of specific plugins (or "artifact")	
Decoupling dependencies	- Outsourcing the maintenance, control and stability of dependencies logic of construction: o override o transitivity o cycle o scope…	
Directing norms	- Lifecycle transverse urbanization by inheritance (disposed aggregated, managed, combined ...), reflecting the company model (organization hierarchical metadata policies): o company o business unit o project o team…	

Continuous Integration

Global Subject



Supported Features

Standard fulfillment	- Continuous integration in a Maven way across state construction notifications through a traceability requirement and/or requirement of immediate responsibility: o notification of success, warning... o notification in case of error, failure...	
Environments scalability	- Consolidation of release following propagation model in target environments: o development o integration o qualification o production...	

Changes availability	- Propagation of changes resulting from a complete automatic construction: o daily o nightly o weekly…	
Build efficiency	- Parallel automatic construction to satisfy a requirement of productivity depending on the project structure and the compatibility of ecosystems	
Developments efficiency	- Automatic incremental construction to satisfy a requirement for agility based project structure and compatibility of ecosystems	