COLLABORATIVE CONSTRUCTION

Collaborative construction is a set of practices that promotes team problem solving, group ownership of code, and frequent informal reviews to improve construction quality and productivity.

Main Benefits	Improves code quality through team problem solving and frequent review.
	May increase productivity through staffing flexibility, improved coordination, reduced planning and tracking overhead, technical knowledge transfer, coaching of junior staff, risk identification, and team problem solving.
	Can improve project cohesion, visibility, and morale.
Keys to Success	Cross-trained construction staff, an open culture, management support, and strong revision control, build, unit test, and smoke test practices.
When to Use	Can be used during construction of almost any project.
Main Risks	Inefficiencies can be introduced on projects that have areas which require significant technical specialization. A balance needs to be struck between having everyone work on everything and leveraging individual strengths.

Overview

Collaborative construction is a set of practices that promotes a team approach to detailed construction activities and group ownership of a system. With collaborative construction everyone knows something about every piece of the system. Team members build, modify, or fix portions of the system as necessary to complete tasks, not along strict lines of area ownership.

The four main techniques behind collaborative construction are listed below. These are not exclusive, and other team oriented practices can provide benefit as well.

- Detailed construction tasks such as low level design, coding, and building content or configuration data are shared. Individuals will often interleave work in the same files.

- When staffing tasks, assign a group of 2-4 team members working together to execute work packages.

- Creation and maintenance responsibility for areas is frequently swapped between team members, so existing code gets lots of informal review by engineers as they add, extend, or fix it. Ideally, as a project evolves, it becomes impossible to identify a single "owner" of a particular area.

- Encourage frequent team problem solving and informal reviews. These can be scheduled meetings like previews and walkthroughs, or hallway desk checks (e.g., "hey, can you take a quick look at this?").

These practices improve the quality of the code produced by ensuring ongoing review of code throughout the project. Strong peer encouragement for good construction practices is provided because of the immediacy of someone else looking at an maintaining code that is being created.

Productivity can be improved through better communication, coordination, cross-training, and risk identification. Project management overhead can be reduced in some cases, because detailed coordination of work tasks can occur informally at the individual level.

Although group ownership is encouraged, optimal use of collaborative construction requires striking a balance between generalization and specialization of individual engineers. It is expected that each engineer will still have specializations and spend the majority of their time in certain areas. The goal of collaborative construction is to promote cross pollination on a project to increase quality, flexibility, and long-term productivity, but this needs to be balanced against the expense of short-term productivity. How this balance is struck will be unique for each project; projects should regularly take objective and subjectively measures of whether the current balance is optimal.

CxOne Support

CxOne provides support for collaborative construction with coding standards, design and construction checklists, construction testing, detailed project planning and tracking materials, issue management materials, performance metric materials, and informal quality review materials.

Interactions with other Best Practices

Collaborative construction is especially effective when combined with daily builds and construction testing. It can be combined freely with other best practices.

Collaborative construction overlaps with several principles of Extreme Programming, but collaborative construction does not advocate a prescriptive approach as being necessary to reap benefits. Most notably, pair programming is not seen as part of this best practice. Although it is possible to use pair programming with collaborative construction, Construx recommends that individual engineers maintain their own workstations and perform the majority of individual code writing individually. The benefits from pair programming are more efficiently realized through post-creation review than from looking over each other's shoulders while typing.