

Build Artifact Management Design Supporting Continuous Deployment

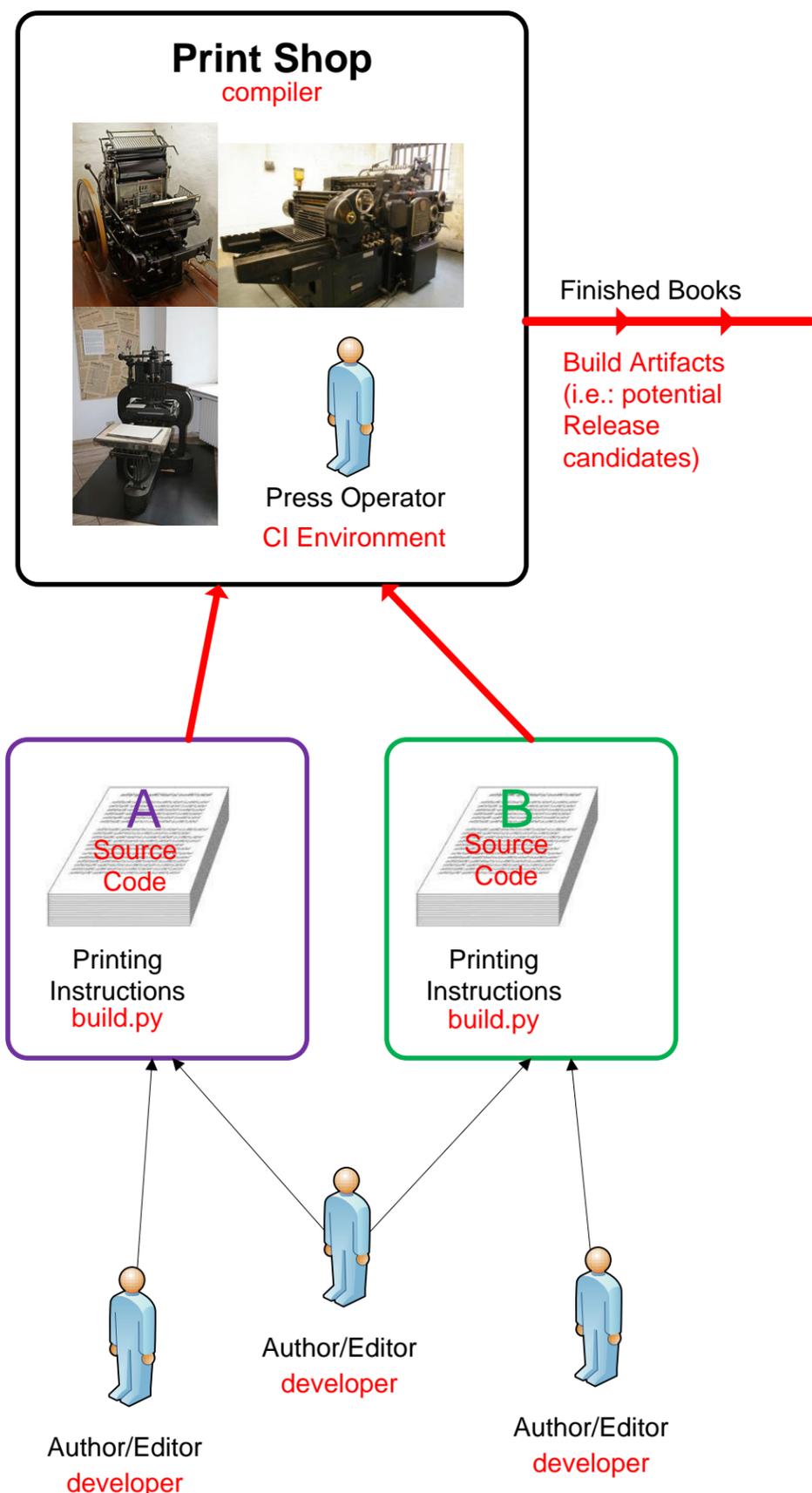
Book Publishing Analogy

Our design for build artifact management is fairly close to library management of physical books. Processes and actors in the physical realm are labeled in black with their analogous build artifact management design pairing labeled in red.

The concept of promoting build artifacts to higher level repositories as they successfully pass through a gauntlet of tests slightly strains the analogy with a physical library. For that part just do what Mr. Rogers would do and pretend to visit the land of make believe.

Key Technical Points

- Source control systems such as ClearCase, Subversion and Git are optimized for managing changes to text files.
- Build artifact repositories such as Artifactory are optimized for managing binary files such flash images, shared object libraries, source zip files, and the like.
- A command line build is expected to obtain source code library dependencies by accessing a build artifact repository.
- A command line build should be capable of publishing to a build artifact repository.
- Every build should have the potential to become a release artifact.
- Since every build has the potential to become a release artifact, every build should have full traceability.
- Virtual repository indices can be amalgamated from any set of other indices with a specified precedence order.
- Please respect sub-system and organizational boundaries when deciding whether to allow a given set of code to leverage build artifacts from lower promotion levels.
- No artifact should be promoted above the promotion level of any of its dependencies.
- Support for code reuse at the build artifact level makes it possible to maintain a single source of truth without resorting to complex source code branch management techniques.
- Although only 3 levels of promotion are shown, every significant milestone a build artifact successfully passes will typically correspond to a promotion level.
- Builds that produce multiple artifacts should ensure all artifacts are published atomically.



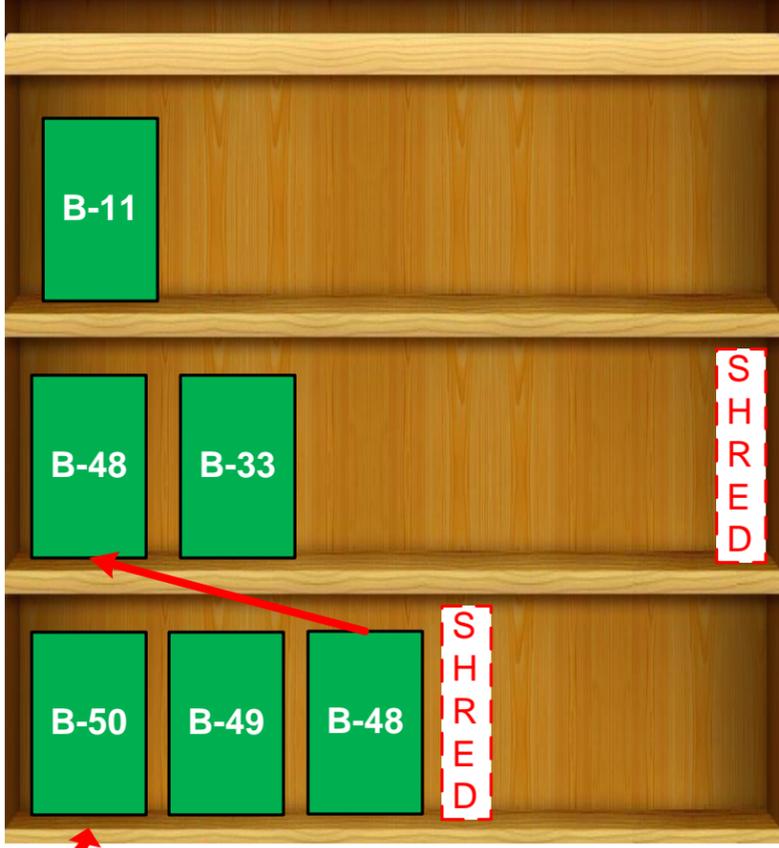
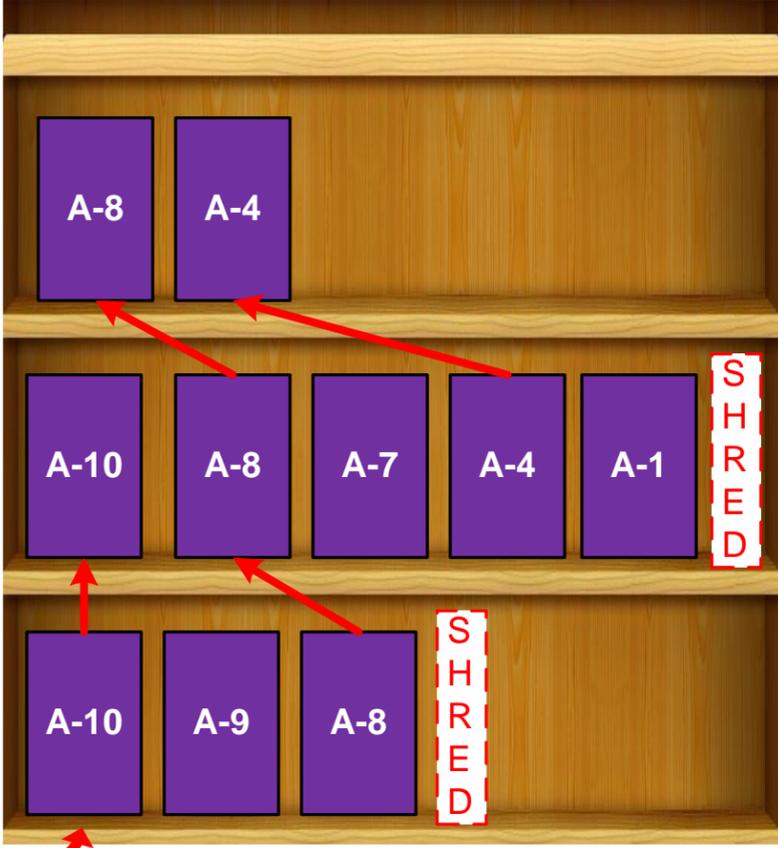
Partner Library
External Build Artifact Repository

Partner Library
External Build Artifact Repository

Inter-Library Loans
Proxied External Build
Artifact Repositories

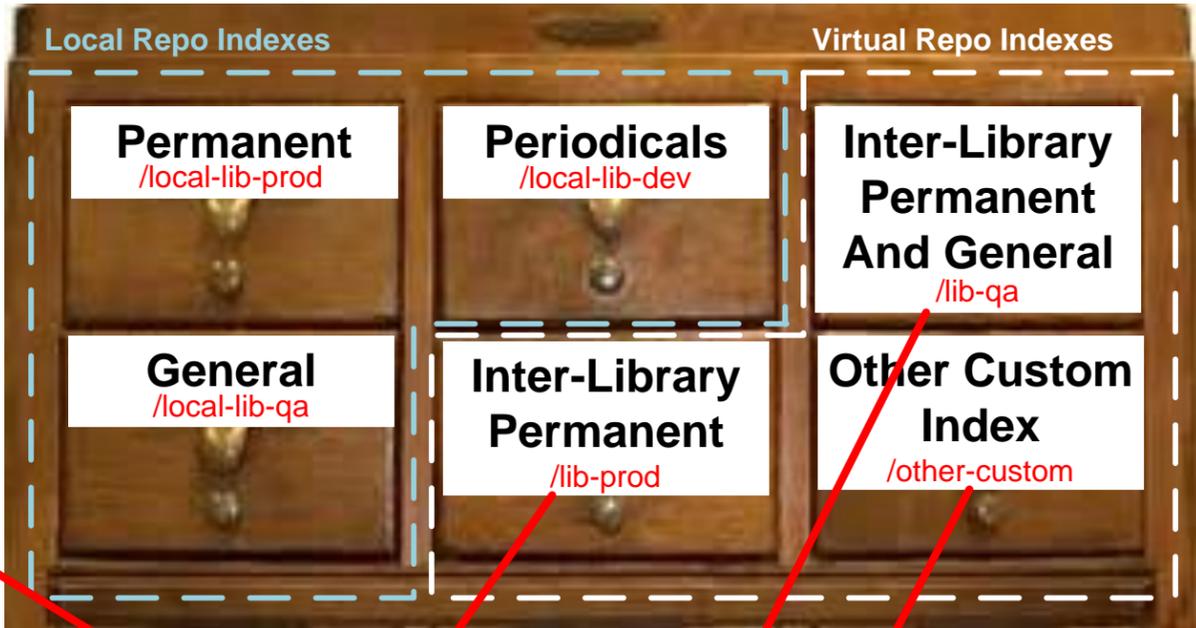
Local Library
Build Artifact Repository Manger

Permanent Collection
/local-lib-prod



Periodicals
/local-lib-dev

Finished Books
Build Artifacts



Normal Library Consumer
Typical Build Artifact Consumer
(e.g.: build systems,
deployment systems,
etc.)

Privileged Library Consumer
Privileged Build Artifact Consumer
(e.g.: build systems,
deployment systems,
etc.)

Book Req./Resp.
Artifact Req./Resp.

Book Req./Resp.
Artifact Req./Resp.

Book Req./Resp.
Artifact Req./Resp.

Book Req./Resp.
Artifact Req./Resp.