CHECKLIST

Subversion to Helix Core Migration Planning Checklist

Migrations from Subversion (SVN) to Helix Core vary in scale and complexity. Many migrations are straightforward and can be completed in hours. For larger and more complex environments, migrations will take longer and a more formal approach is recommended.

Use this checklist to help you plan your migration from SVN to Helix Core.

☐ **SELECT AND TRAIN TRANSITION TEAM**
Your transition team is fundamental to a successful migration. Make sure your transition team includes software developers and your SVN admin. If possible, include someone with SVN to Helix Core migration experience.

☐ **PLAN YOUR BRANCHING STRATEGY**
Determine whether your current branching strategy is appropriate to use in Helix Core. If not, decide what branching strategy you'll use after transitioning to Helix Core.

☐ **SELECT YOUR MIGRATION APPROACH**
Determine what and how much history to import. This requires careful consideration, as you may have different approaches for different products. Migration options include Tips Only Detailed History Import (DHI), and Baseline and Branch Import (BBI).

☐ **DETERMINE YOUR CUTOVER STRATEGY**
Determine if a big bang or phased rollout strategy is more appropriate. The big bang approach cuts over everything at once and is the recommended strategy. If this isn't feasible, perhaps due to large projects or individual team readiness, then the phased rollout strategy is necessary.

☐ **IDENTIFY YOUR TRANSITION SCHEDULE**
Determine the best time to execute the cutover. If you're doing a big bang, this may be over a weekend or when there are fewer projects in process. If you are planning a phased rollout, use the same considerations to prioritize and schedule team cutovers.

☐ **COMPLETE DRY RUNS**
Completing dry runs is essential for success. Dry run everything. Do full dry runs against various stresses and loads, test your backup systems, test integration with your developers' tools, have developers run from their desktops, and so on. Address any data or process snags. And consider a production simulation via pilot and proof of concept.

☐ **GET THE HELIX CORE ENVIRONMENT READY**
Make sure everything migrated completely and the source code is verified in your Helix Core system. Do any necessary retooling on your build systems, such as custom automation that needs to integrate with Helix Core. Verify the build results – make sure you can do a build, complete a build, and run it through your test suites.

☐ **TRAIN USERS**
Do this as close to your cutover date as possible, but with a little bit of leeway in case any new or missed opportunities are brought up as a result of the training. Perforce offers online and class-room-based trainings so your team can quickly leverage the power of Helix Core.

☐ **CUTOVER TO HELIX CORE**
Start enjoying all the benefits of Helix Core.