

Change Management Evaluation Checklist

Functionality

- Overall, does the product feel intuitive?
- Will it support future development activities as well as your development today? Will it support your enterprise CI/CD requirements?
- Does it support all SQL types including packages? Does it recognize the creation order needed for complex relationships of views, services programs, and types, and provide for easy management?
- How does the product handle exceptions to the norm? Is it easy to override or customize?
- How does it handle approval processing and tracking?
- Does it provide source compare and conflict management functionality?
- Does the product appear easy to administer?
- Is audit reporting part of the product, or does it have to be purchased separately?
- Can source be checked out via the IFS or imported from a source repository such as Git or SVN?
- Are you required to learn a new language to use the tools?

Project Management

- Can approval be done by business users without direct access the CM product?
- Discuss your approval and notification requirements to determine if the product will meet them.
- If you use another project/service desk tool, is there currently an interface to the CM product? In most cases, this is done via REST APIs.

IBM i Specific Check Points

- Are there easy ways to manage queries, menus, control tables, and SQL?
- Rollback capabilities are not all created equal; make sure you know what can and cannot be done.
- What are the limits on the number of test levels between development and production?
- Does it support pre/post install commands per object and per deployment?
- What type of cross-referencing is available within the product: Field level or Object level?
- Can you link native and non-native deployments?
- Can source be compiled on target systems?
- Can source be stored in a Git or SVN repository? Can it automatically handle imports and commits to Git or SVN?
- Are RDi and VS Code accessible from directly within the product UI?

Change Management Evaluation Checklist

Software Vendor Interfaces

- Do vendor interfaces incur additional licensing cost?
- Who wrote the interface and how is it supported? Many older 3rd party interfaces have minimal or no support.
- Does the CM product provide REST APIs to support access?
- Is the vendor or a 3rd party responsible for bug fixes and upgrades for the interface?

About the Vendor

- How long has the software provider been in business?
- How often are new releases generated? This indicates the vendor's commitment to the product.
- Take a look at the vendor's major release notes. Are they providing new functionality or just bug fixes?

Ask for references that use the product in much the same way as you will. Call them and ask lots of questions:

- How did the implementation go? Request timeline.
 - How well do interfaces function?
 - How often do they have to contact Tech Support? How quickly are problems resolved?
 - How does the vendor respond to enhancement requests?
 - If they migrated from another CM tool, how smooth was the transition? What historical information were they able to keep? How did work in progress move into the new product?
- Is the vendor owned by the original investors or by a technology holding company? Investor groups and holding companies provide financing, but keep in mind that R&D is usually low priority.

Pricing and Availability

- Is the vendor willing to discuss your environment and provide out-of-the-box solutions?
- Must additional modules be purchased to support your environment? Evaluate total cost.
- Bells and Whistles – Do you need all the "extras" they are proposing?
- Are there additional license costs for High Availability and Disaster Recovery systems?

The Software License

- Does the license protect your investment with a money-back guarantee period?
- Must you pay maintenance for the product to continue to work? Watch for kill switches or "back doors" programmed into the application for purposes of disabling the software.