



# CMO<sup>ne</sup> – Das Change Management System für i5, iSeries und AS/400

## Technische Übersicht

- **TOP1:** WDSC und RDi ready. PlugIn's für beide graphischen Entwicklungswerkzeuge als Nachfolger der klassischen Green-Screen-Umgebung PDM stehen in der Version **CMO<sup>ne</sup>** Next Generation (NG) zum Download bereit.
- **TOP2:** Mit der Version **CMO<sup>ne</sup>** NG haben Sie eine zentrale Entwicklungsumgebung (IDE), die Ihnen in graphischer Form alle **CMO<sup>ne</sup>** Funktionalitäten zur Navigation bereitstellt.
- **TOP3:** In der Version **CMO<sup>ne</sup>** NG stehen Ihnen unter RDi moderne Editoren wie LPEX und Screen Designer zur Verfügung.
- **TOP4:** Über das RDi Plugin **CMO<sup>ne</sup>** NG erlangen Sie mit dem graphischen Anwendungsdiagramm schnell einen Überblick Ihres komplexen Sourcecodes und erkennen u.a. die Verschachtelung von Routinen bzw. Prozeduren. Eine weitere Hilfe, sich in fremden Sourcen schnell zu Recht zu finden, ist die Gliederungsübersicht.
- **TOP5:** CheckOut-, CheckIn-Verwaltung zu Quellen und Objekten.
- **TOP6:** Auf dem Arbeitsblatt befinden Sie alle notwendigen Objekte zu einer Aufgabe, so dass Sie immer den Überblick behalten.
- **TOP7:** Spezielle Unterstützung konkurrierender gleichzeitiger Bearbeitung ein und derselben Quelle, ein und desselben Objekts durch Mehrfach-CheckOut-Management.
- **TOP8:** Projekt- und Aufgabendokumentation, so dass all Ihre Entwicklungen schriftlich festgehalten werden können.
- **TOP9:** Unterstützung sämtlicher nativer OS/400 Objektarten - \*MODULE, \*PGM, \*SRVPGM, \*FILE, \*MSGF, \*DTAQ ...
- **TOP10:** Unterstützung sämtlicher Compiler, welche native OS/400 Objektarten erzeugen – RPG, COBOL, C, C++, CL ...

- **TOP11:** Handling von ILE (Integrated Language Environment) Strukturen - Modul, Programm, Serviceprogramm ...
- **TOP12:** Benutzeroberfläche in Deutsch und in Englisch.
- **TOP13:** Spezielle Unterstützung von SQL-DDL (Structured Query Language – Data Definition Language) – Table, Index, View ...
- **TOP14:** Berücksichtigung von Triggern und Constraints.
- **TOP15:** Integritätsberichte mit folgenden Informationen:
  - Quelle vorhanden, aber das dazugehörige Objekt fehlt.
  - Objekt vorhanden, aber die dazugehörige Quelle fehlt.
- **TOP16:** Verwaltung von PTF`s analog IBM-Handling (Einzel-PTF`s, Kummulative PTF`s, Notwendige PTF`s, Ersetzte PTF`s und Begleitschreiben).
- **TOP17:** Versionsmanagement, so dass unterschiedliche Versionen einer Anwendung verwaltet werden können.
- **TOP18:** Durch vielfältige Exit-Points ist es möglich, eigene Programme in den **CMOne**-Ablauf zu integrieren und dadurch auch individuelle Logik abzubilden.
- **TOP19:** Spezielle Unterstützung nicht mandantenfähiger Software, bei denen die Mandantenfähigkeit über die Verteilung der Daten in eigenen Datenbibliotheken pro Mandant erzeugt wird. Bei Änderung eines in **CMOne** als Datenobjekt definierten Objekts (z. B. physische oder logische Datei) wird dieses Objekt in allen Datenbibliotheken automatisch installiert.
- **TOP20:** Zentrale Definition von Eignern und Berechtigungen zu Objekten, mit denen Objekte erstellt werden sollen. Dadurch gibt es keine Probleme mehr mit Programmabbrüchen durch falsche Berechtigungen beim Betrieb der Anwendung.
- **TOP21:** Definition der Bibliotheken, in denen an Quellen und Objekten zu einer Arbeitsanforderung gearbeitet wird. Auf den Ebenen Anwendung, Projekt, Aufgabe und Benutzer möglich. Dadurch kann die Wahl der Entwicklungsumgebung sehr flexibel gestaltet werden.

- **TOP22:** Verwaltung von Arbeitsanforderungen über die Ebenen Projekt, Aufgabe und Arbeitsblatt.
- **TOP23:** Verwaltung von Befehlen zur Kompilierung von Objekten in Form von:
  - Vorlaufbefehlen, die vor dem eigentlichen Erstellungsbefehl aufgeführt werden,
  - Nachlaufbefehlen, die nach dem eigentlichen Erstellungsbefehl ausgeführt werden und den eigentlichen
  - Erstellungsbefehlen, mit denen Objekte letztendlich kompiliert werden.
- **TOP24:** Spezielle Unterstützung mehrsprachiger Anwendungen, wenn die Mehrsprachigkeit über Nachrichtendateien erreicht wird.
- **TOP25:** Integriertes Objektverzeichnis zu allen in einer Anwendung existierenden Quellen und Objekten.
- **TOP26:** Umfangreiche integrierte Objektreferenzen. Folgende Informationen werden geliefert:
  - Alle Informationen, die der Befehl DSPPGMREF zu einem Objekt liefert.
  - Eingabedateien in Query`s.
  - Externe Datenstrukturen in RPG III und RPG IV Quellen.
  - Physische Dateien, auf denen logische Dateien basieren.
  - Module in Programmen und Serviceprogrammen.
  - Module und Serviceprogramme in Binderverzeichnissen.
  - Copy-Strecken in RPG III und RPG IV Quellen (/COPY).
  - Befehlsverarbeitungsprogramme (CPP) in Befehlen.
  - Alle Objekte, auf die in DDS-Quellen (PF, LF, DSPF, PRTF ...) über ein DDS Schlüsselwort (PFILE, FORMAT, MSGCON, TRNTBL ...) Bezug genommen wird.
  - Beziehungen zwischen Objekten, die hier nicht genannt sind, können in **CMOne** als benutzerdefinierte Objektreferenz eingetragen werden. Alternativ hierzu kann noch über die **CMOne** Direktive %REF% direkt im Quellcode die Beziehung angegeben werden. Derart definierte Beziehungen werden dann von **CMOne** als Objektreferenzen erkannt.

- **TOP27:** Umfangreiche integrierte Feldreferenzen. Folgende Informationen werden geliefert:
  - Dateien (Physische Dateien, Logische Dateien, Bildschirmdateien ...).
  - Importe von Felddefinitionen (z.B. über die DDS-Schlüsselwörter REF, REFFLD, RENAME, CONCAT ...).
  - Exporte von Felddefinitionen (Das sind die Felder in Dateien, auf die in Programmen und Modulen zugegriffen wird).
  - Programme und Module.
  - Importe von Felddefinitionen (Das sind die Felder in Dateien, auf die in Programmen und Modulen zugegriffen wird, über externe Dateibeschreibungen, externe Datenstrukturen und SQL-Befehlen in ILE RPG Quellen, sowie über den DCLF Befehl in ILE CL Quellen).
  
- **TOP28:** Über die integrierten Feldreferenzen ist es u.A. jederzeit möglich sich die Informationen zu beschaffen, welche Programme und Module ein bestimmtes Feld in einer physischen Datei direkt oder indirekt verwenden. (Direkt bei Verwendung der physischen Datei, indirekt bei Verwendung der z.B. logischen Datei).
  
- **TOP29:** Umfangreiche integrierte Symbolreferenzen. Folgende Informationen werden geliefert:
  - Datenimporte und Exporte (z.B. Felddefinitionen in ILE RPG Quellen, bei denen die Schlüsselwörter IMPORT und EXPORT angegeben werden).
  - Prozedurenimporte und Exporte (z.B. Prozedurendefinitionen in ILE RPG Quellen, bei denen die Schlüsselwörter IMPORT und EXPORT angegeben werden).
  - Wer 100% ILE programmieren will, kann hierauf nicht verzichten.
  
- **TOP30:** Automatische Rekompilierung abhängiger Objekte, wenn entsprechende Objekte geändert werden. Beispiel: Wird eine physische Datei geändert, dann werden automatisch alle logischen Dateien, welche auf der physischen Datei basieren rekompiliert. Weiterhin werden alle Module, Programme und Serviceprogramme rekompiliert, welche die physische Datei bzw. eine auf der physischen Datei basierende logische Datei verwenden.

- **TOP31:** Erfüllung aller Revisionsanforderungen wie z.B. SOX, Basel II.
- **TOP32:** Archivierung alter Stände zu Quellen und Objekten im Online-Archiv (Platte) bzw. Offline-Archiv (Band, DVD ...).
- **TOP33:** Verteilung von Quellen und/oder Objekten auf beliebig viele Produktionssysteme via TCP/IP, SNADS oder Band.
- **TOP34:** Vollautomatische Installation von Quellen und/oder Objekten auf den Produktionssystemen mit Übermittlung des aktuellen Installationsstatus an das Entwicklungssystem.
- **TOP35:** Bei der Installation physischer Dateien wird folgendes Verfahren durchlaufen:
  - Alte Datei retten.
  - Neue Datei erstellen.
  - Teildateien, die in der alten Datei vorhanden sind werden in der neuen Datei angelegt.
  - Daten werden je Teildatei von der alten Datei zur neuen Datei kopiert.
  - Constraints der neuen Datei hinzufügen.
  - Trigger der neuen Datei hinzufügen.
- **TOP36:** Über **CMOne** installierte Änderungen an Quellen und Objekten können wieder auf den alten Stand zurückgesetzt werden, insofern diese Änderungen Probleme verursachen wie z.B. Programmfehler (Rollback).
- **TOP37:** Änderungen an Quellen und Objekten können vor der eigentlichen Installation auf dem Produktionssystem in einer Testumgebung installiert werden (Special-Form Handling). Nach Installation der Quellen und/oder Objekte in der Produktionsumgebung werden diese in der Testumgebung gelöscht.
- **TOP38:** Verwaltung sämtlicher Objekttypen und Ersatzvariablen mit **CMOne** eigenem Typcode.
- **TOP39:** Verwaltung von Standards zum Formlauf in der Entwicklungs- und Produktionsumgebung.
- **TOP40:** Verwaltung von Benutzern mit Berechtigungen und Sprache.



- **TOP41:** Vollautomatische Funktion zum Einmischen von Sourceänderungen beim Mehrfach-Checkout.
- **TOP42:** Über die Funktion ‚Volltextsuche‘ gibt es in allen entsprechenden Sichten von **CMOne** NG die Möglichkeit, komfortabel Quellenteildateien über eine Volltextsuche zu durchsuchen.
- **TOP43:** Mit der Funktion ‚Vergleichen‘ ist ein Komfortables Vergleichen von verschiedenen Quellenständen (Anwendung, PTF, Objekthistorie) möglich.

*Diese Kurzbeschreibung gibt nur einen groben Überblick über die Funktionalitäten, die **CMOne** bietet. Im Detail gibt es weit mehr, als hier aufgeführt. Lassen Sie sich die Funktionalitäten mal live vorführen...*