

1 COM

Distribuering af .dll-filer

- Førhen brugte man .obj-filer. Disse fylder dog meget, og man kan ikke ændre i dem (kun udviklerne kan). Disse kan dog linkes til en .exe-fil.
- I stedet kan man tage en .dll-fil, som kun tager det nødvendige fra en .obj-fil.
- En .dll bliver kun oprettet én gang og alle applikationer der bruger den, bliver opdateret.
- C++ er dog kun standardiseret på Source-niveau og ikke på Binary-niveau - dette kan medføre problemer.
- Laver man en opdatering på en .dll-fil, skal de programmer der bruger dem recompileres

Måden man omgår dette på:

- Man laver en handler, der blot indeholder en pointer til det konkrete objekts funktioner. På den måde udvider .dll-filen sig ikke.
- Man laver en abstrakt base-class (pure virtual). Clienten har kun brug for interfacet. Her skal man dog have et .dll, der har en 'create new concrete object' samt en 'delete concrete object'.

1.1 Resource Management

- Hold styr på, hvor mange der bruger et object. Når der ikke er flere, kan det nedlægges.

1.2 Goals of COM

- Binary encapsulation: Clienter skal ikke recompilere kode
- Binary compatibility: Filer kan bruges påtvær at platforme

1.3 Principles

- Alt info går igennem et interface
- Ret aldrig i et udgivet interface

1.4 Arkitektur

Bruges lokalt, men er så godt som helt dødt på remote.

- Client spørger COM Runtime (CR) efter et objekt.
- CR spørger server efter pointer, der returneres
- Client bruger pointer til server.

COM bruger Registreringsdatabasen. Her ligger alle informationerne om COM-elementerne.