F OLE — COM & DCOM

But the darkest of dark sides is a thing Microsoft hastily concoted called OLE ... OLE is catching on like e-mail at Al Gore's house, but unfortunately it is the deadliest possible blow to OOT — mainly because it isn't object-oriented, it isn't fully specified, it's inferior to OpenDoc, and there is little anyone can do to stop it. OLE will be synonymous with objects.
(Ted Lewis, IEEE Computer Magazine, Dez. 1994)

F.1 References

Wan+97. Y. M. Wang et al.: DCOM and CORBA Side by Side, Step By Step, and Layer by Layer, http://akpublic.research.att.com/~ymwang/papers/HTML/DCOMnCORBA/S.html, C++ Report Magazine, September 1997.

MS97. The Component Object Model Specification, http://www.microsoft.com/oledev/olecom/title.htm.

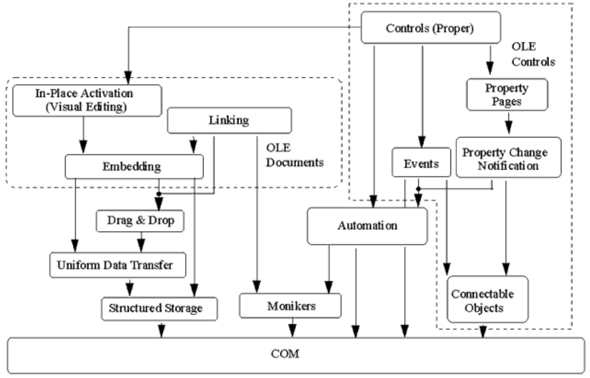
BrKi96. N. Brown, C. Kindel, Distributed Component Object Model Protocol -- DCOM/1.0, http://www.microsoft.com/oledev/olecom/draft-brown-dcom-v1-spec-01.txt.

Kirt97. M. Kirtland, The COM+ Programming Model Makes it Easy to Write Components in Any Language, Microsoft Systems Journal, December 1997, http://www.microsoft.com/msj/1297/complus2/complus2.htm.

F.2 Overview

- OLE: Object Linking and Embedding
  - ➥ Microsoft's "standard" for collaboration of software components
    - ➤ Text processor & spreadsheet calculator & drawing tool
    - ➤ Spreadsheet calculator & database
    - ➤ Scripts with applications
    - ➤ ...
- COM: Component Object Model
  - ➥ An Object Request Broker for a single-machine environment
    - ➤ Object Bus & Object Services
- DCOM: Distributed COM
- ActiveX: COM enabled for the Internet (what ever that means)

F.3 OLE Architecture



from: Microsoft: COM Techncical Overview

F.4 COM IDL & COM ODL

1 IDL

- Interface definition language, used to create
  - ➤ client proxies
  - ➤ server stubs
  - ➤ C code for parameter marshalling
- MIDL Compiler
  - ➤ stub generator
- Comparison to CORBA IDL
  - ◆ no language binding
    - ➤ binary "standard" for vtable with pointers to methods
  - ◆ no multiple inheritance on interfaces

2 ODL

- Object description language
  - ◆ used to describe object interfaces for
    - ➥ Type Library (= Interface Repository)
- MKTYPLIB
  - ◆ utility to generate
    - ➤ interface metadate for type library
    - ➤ header file with function prototypes

F.5 COM: OLE's Object Bus

1 !!! "COM objects" are not objects

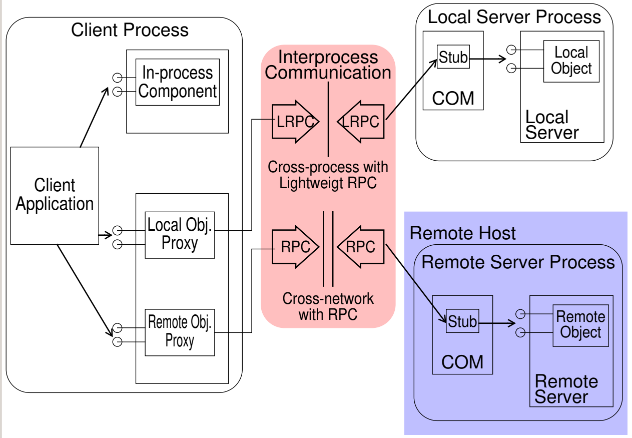
In COM, an object is a piece of compiled code that provides some service to the rest of the system. To avoid confusion, it is probably best to refer to an object used in COM as a COM component or simply as a component. This avoids confusing COM components with source-code OOP objects such as those defined in C++. (Microsoft: COM Tehnical Overview)

- As CORBA, COM separates the object interface from its implementation
  - ➤ separate interface description in IDL
  - ➤ one COM object can support multiple interfaces
- COM interfaces are interfaces to services, not to "real objects"
  - ◆ COM objects have no unique id
    - ➥ COM objects are collections of functions
    - ➥ COM objects do not maintain state

2 COM server

- Like CORBA sever
  - ◆ creation and execution environment for COM objects
  - ◆ object creation via factory interface
- Flavors
  - ➤ In-process servers
  - ➤ Local servers
  - ➤ Remote servers

3 COM Client/Server Interaction



F.6 COM/OLE Object Services

- Interface negotiations
  - ➤ client initially gets only primary interface
  - ➤ client may request reference to another interface
  - ➤ if supported the target object returns reference to that interface
- Life cycle management
  - ➤ factories
  - ➤ reference counting & deletion of unused objects
- Component licensing
  - ➤ object instantiation depending on license files
- Event service (connectable objects)

F.7 OLE Automation

- Dynamic Invocation Interface = OLE Automation
  - • OLE Automation allows a single program to control automation servers residing in many applications
  - • automation server: interface objects of a scriptable application

F.8 OLE Uniform Data Transfer

- Clipboard transfer
- Drag&drop
- Links
  - ➤ notification mechanism

F.9 OLE: Structured Storage

➥ Persistence service

- Compound Files (future MS file system?)
  - ➤ transactional storage
- Persistent objects
  - ➤ can read and write themselves to storage
- Monikers
  - ◆ persistent intelligent names
    - ➤ naming service + interface to the objects behind it
  - ◆ implement name-related interfaces
  - ◆ names for files, tables, table-cells, ...
  - ◆ work-around for persistent objects
    (COM provides references to interfaces of services, not to objects, solution: moniker can instantiate an object an reestablish it's former state)

F.10 DCOM

- Distributed extension to COM
- Remote method invocation based on DCE RPC
- Open technology
  - ➤ DCOM submitted to IETF to become an Internet standard
  - ➤ Available for Windows NT 4.0, Windows 95, MacOS
  - ➤ Software AG builds port for Solaris