

**Symbols**

\_duplicate 20-7, 20-9  
 \_major 19-27  
 \_narrow 20-9, 20-60  
 \_nil 20-10  
 \_ptr field accessor 20-27  
 \_tie\_A class 20-94  
 \_var 20-7  
 'SIZE 23-20

**A**

A\_ptr 20-6, 20-7, 20-117  
 A\_var 20-6, 20-7  
 aBool 21-9  
 abstract base class 20-6  
 access function 20-32  
 Accessor Functions 22-47  
 aCORBAObject 21-19, 21-22  
 activation 1-8, 1  
 Ada Implementation Requirements 23-2  
 Ada package 23-2  
 addArg 21-19  
 aDiscriminator instance method 21-15  
 aggregate type 20-62  
 alias 20-44  
 AliasDef  
   OMG IDL for 8-22  
 alignment 13-10  
 Alternative Mappings for C++ 21-24, 23-45, 23-65  
 ANSI COBOL 85 language 22-2  
 ANSI/ISO C++ standardization committees 20-3, 20-118  
 Any 23-29  
 Any class  
   helper types 20-52  
 any class 20-105  
 any type 3-25, 5-3, 13-17, 16-10, 16-38, 19-11, 20-46, 21-4, 21-12  
   conversion of typed values into 20-46  
 Any values  
   dynamic management overview 7-2  
 Any\_var 20-57  
 application object xxv  
 ARef 20-6  
 Arguments, Passing 23-36  
 Arithmetic Types 23-21  
 array  
   sample mapping to OLE collection 17-49  
   syntax of 3-29  
 array slice 20-33  
 Array\_forany 20-43  
 Array\_var 20-43  
 ArrayDef  
   OMG IDL for 8-25  
 Arrays 23-26  
 assignment operator 20-21, 20-31  
 attribute 1  
   defined 1-7  
   mapped to OLE 17-4  
   mapping to COM 16-23  
   mapping to OLE Automation 15-10  
   mapping to programming languages 19-5  
 attribute declaration

  syntax of 3-33  
 Attribute\_Def  
   OMG IDL for 8-26  
 Attributes 23-3, 23-35, 23-44  
 Attributes, Server Side 23-44  
 Automation View Dual interface, default name 15-31  
 Automation View interface 17-2, 17-16  
   non-dual 17-36  
 Automation View interface class id 15-32  
 Automation View interface, default name 15-30  
 Automation View interface, default tag 15-30

**B**

BAD\_PARAM exception 20-65  
 BadCall exception 19-28  
 base exception class 20-58  
 base interface 3-16  
 base interface type 20-8  
 basic data types  
   and different platforms 20-15  
   mapped from OMG IDL to C 19-11  
   mapped from OMG IDL to C++ 20-15  
   mapped to programming languages 19-3  
 basic object adapter 17-37, 20-91  
   mapped to C 19-42  
 big-endian 13-7  
 binding 15-20  
 BindingIterator interface 17-60  
 Boolean 23-21  
 boolean 17-60  
 boolean is\_a operation  
   OMG PIDL for 4-6  
 boolean type 20-15, 20-16, 21-12  
 boolean types 3-25, 13-10, 20-15  
   mapped to C 19-11  
 bridge  
   architecture of inter-ORB 11-2  
   in networks 11-11  
   inter-domain 11-9  
   inter-ORB 10-2, 10-5, 11-6  
   locality 15-33  
 bridging techniques 11-8

**C**

C  
 \_major field 19-27  
 and is\_nil operation 19-7  
 any type 19-11  
 attribute mapping examples 19-8  
 BadCall exception 19-28  
 basic data type mapping 19-11  
 boolean types 19-11  
 global name 19-5  
 inheritance of operations 19-8  
 ORB initialization 19-44  
 signature of Dynamic Implementation Routine 19-42  
 underscore characters in mapping 19-9  
 C++ 20-61  
   \_duplicate 20-7, 20-9  
   \_narrow 20-9, 20-60  
   \_nil 20-10

- `_ptr` field accessor 20-27
- `_tie_A` class 20-94
- `_var` 20-7
- `A*` 20-7
- `A_ptr` 20-6, 20-117
- `A_var` 20-6
- abstract base class 20-6
- aggregate types 20-62
- alias 20-44
- and struct 20-27
- Any class interface 20-52
- any type 20-56
- `Any_var` 20-57
- `ARef` 20-6
- arglist 19-45, 20-86
- arithmetic operations 20-7
- array 20-41
- array slice 20-33
- `Array_forany` 20-43
- `Array_var` 20-43
- assignment operator 20-31
- automatic release 20-64
- basic data type mapping 20-15
- boolean type 20-16
- catch clause 20-60
- char type 20-16
- `char*` 20-17
- `CompletionStatus` 20-59
- constant 20-13
- Context interface, OMG PIDL for 20-80
- conversion to `void*` 20-7
- CORBA
  - Object 20-69
- CORBA Boolean 20-15
- CORBA Char 20-15
- CORBA Double 20-15
- CORBA Float 20-15
- CORBA long 20-15
- CORBA namespace 20-103
- CORBA Octet 20-15
- CORBA Short 20-15
- CORBA ULong 20-15
- CORBA UShort 20-15
- delete 20-23
- deleting old character data 20-64
- discriminant 20-31
- Double 20-16
- duplicate 20-8
- `dynamic_cast<T*>` 20-61
- enumeration type 20-16
- Environment 20-117
- Environment interface, OMG PIDL for 20-71
- extraction of values 20-50
- Float 20-16
- function overloading 20-46
- generated class 20-6
- implicit release 20-7
- implicit widening 20-7
- insertion of a string type 20-48
- insertion of arrays, type-safe 20-48
- `is_nil` operation 20-8
- keywords 20-5, 20-118
- keywords, list of 20-118
- left-shift-assign operator 20-47
- Long 20-16
- mapped for non-exception handling environments 20-116
- mapped for non-namespace environments 20-116
- mapped to ORB initialization operations 20-85
- mapping compatability to C 20-4
- modifier function 20-33
- NamedValue interface, OMG PIDL for 20-73
- namespace 20-3, 20-5
- nested constant 20-13
- NVList interface, OMG PIDL for 20-74
- NVList type 20-100
- Object interface, OMG PIDL for 20-87
- object reference variable type 20-7
- `Object_ptr` 20-8
- `Object_var` 20-8
- octet type 20-16
- oneway 20-61
- operation-specific arguments 20-62
- operator< 20-47
- operator-> 20-23
- operator>>= 20-50
- operator[] 20-38
- ORB interface, OMG PIDL for 20-83
- `ORB_init` operation 19-45, 20-86
- overloaded subscript operator 20-38
- parameter passing 20-62
- pointer type 20-7
- portability of implementations 20-16
- primitive type 20-103
- read-write access 20-33
- relational operations 20-7
- release operation 20-8
- release parameter 20-37
- replace function 20-56
- Request interface, OMG PIDL for 20-77
- returning or passing null pointers 20-64
- right-shift-operator 20-50
- run time type information 20-61
- sample COM mapping 16-17
- sample interface mapping 20-11
- sequence types 20-35
- server 20-88
- set function 20-61
- setting union value 20-32
- `sizeof(T)` 20-5
- skeleton class 20-93
- slice 20-42
- split allocation 20-63
- storage 20-64
- string union members 20-34
- `String_var` 20-17
- structured types 20-21
- SystemException 20-59
- `T *data` constructor 20-37
- `T_ptr*` 20-38
- `T_var` 20-21, 20-118
- template 20-93
- throw exception 20-97

- tie class 20-94
- type function 20-57
- TypeCode 20-46
- TypeCode and value, mismatched 20-46
- TypeCode\_ptr 20-57, 20-81
- typedef 20-44
- ULong 20-16
- underscore 20-89
- union members 20-31
- unsafe operations 20-56
- untyped value 20-56
- UserException 20-58
- UShort 20-16
- using statement 20-4, 20-5
- value function 20-57
- void\* 20-57
- Calling Convention 23-5
- catch clause 20-60
- caught 20-60
- CDR 13-4
  - features of 13-3
- char type 3-24, 20-15, 20-16
- char\* 20-17
- char\*\* 20-38
- client 2-7
- CloseConnection 13-31
- CLSID 15-32, 16-44
- COBOL language mapping 22-2
- COM
  - described 15-4
- COM View interface, default name 15-30
- COM View interface, default tag 15-29
- Comments 23-11
- Common Data Representation
  - see CDR
- Common Facilities xxv
- CompletionStatus 20-59
- compliance xxvi
- component
  - tags for A-1
- Component Object Model
  - see COM 15-4
- ConnectionPoint Service 17-52
- constant 20-14
- constant declaration
  - syntax of 3-18
- Constant Expressions 23-8
- Constants 23-27
- constants
  - mapping to programming languages 19-3
- constructed data types 13-10
  - mapping to programming languages 19-3
- Contained interface
  - OMG IDL for 8-11
- Container interface 8-9
  - OMG IDL for 8-13
- containment 11-6
- Context 23-39
- Context interface
  - OMG PIDL for 20-80
- context object 5-13
- copy constructor 20-21
- CORBA
  - Any values
    - dynamic creation of 7-14
    - dynamic interpretation 7-15
  - contributors xxix
  - core xxvii
  - documentation set xxv
  - general language mapping requirements 19-2
  - getResponse instance method 21-20
  - interoperability xxvii
  - namespace 20-103
  - Object 20-69
  - object references and request level bridging 12-6
  - CORBA module
    - C++ definitions for 20-103
    - Context interface 5-15
    - description of 3-34
    - NVList interface 5-11
    - object class 20-8
    - Request interface 5-5
    - types defined by 5-1
  - CORBA package 23-12
  - CORBA\_free 5-4
  - CORBA-alloc 22-4
  - CORBAComposite interface 16-50
  - CORBAConstants 21-10, 21-12, 21-17, 21-18
  - CORBAContext protocol 21-20, 21-22
  - CORBAEnum protocol 21-12
  - CORBAExceptionEvent 21-17
  - CORBAExceptionEvent protocol 21-16
  - CORBAExceptionValue protocol 21-17, 21-19
  - CORBA-fre 22-4
  - CORBANamedValue protocol 21-22, 21-23
  - CORBAObject protocol 21-21
  - CORBAORB protocol 21-21
  - CORBAParameter 21-8
  - corbaRaise message 21-17
  - CORBARequest protocol 21-19
  - CORBA-string-get and CORBA-string-set 22-48
  - CORBAUnion protocol 21-15
  - CORBA-wstring-get & CORBA-wstring-set 22-49
  - core, compliance xxvi
  - CosNaming interface 17-56
  - create\_list operation 5-2
  - create\_request operation 4-4
  - createRequest 21-9
  - CreateType method 17-28
- D**
  - data type
    - basic OMG IDL 3-23–3-25
    - constructed OMG IDL 3-25–3-27
    - constructs for OMG IDL 3-22
    - native 3-23
    - OMG IDL template 3-27–3-28
  - DCE 10-1, 16-1
  - DCE CIOP
    - pipe interface, DCE IDL for 14-6

- DCE CIOP module
  - OMG IDL for 14-25
- DCE ESIOP 11-21
  - see also DCE CIOP
- DCE UUID 15-17
- DCE-CIOP
  - storage in IOR 14-5
- DCORBATypeCode interface 17-24
- DCORBAUnion interface 17-22
- DCORBAUserException interface 17-31
- deactivation 1-8
- derived interface 3-16
- DICORBAAny interface 15-27, 17-24
- DICORBAFactory interface 15-24, 17-26, 17-27
- DICORBAStruct interface 17-21
- DICORBASystemException interface 17-33
- DICORBAUnion interface 17-22
- DICORBAUserException interface 17-31
- Dictionary 21-18
- DIForeignComplexType interface 17-20
- discriminant 20-31
- discriminator instance method 21-15
- domain 11-2
  - architecture 11-5
  - containment 11-6
  - federation 11-6
  - naming objects for multiple 11-12
  - object references 11-12
  - object referencing for 11-12–11-14
  - security 12-4
- double 21-11
- double type 20-15
- Dual interface 15-12, 17-4
- duplicate 20-9
- duplicate operation 20-8
- Dynamic Implementation Routine
  - C signature 19-42
  - mapped to C 19-42
- Dynamic Invocation interface 16-28, 17-37
  - overview of 2-4, 2-9
  - parameters 5-2
  - request level bridging 12-6
  - request routines 5-5
- Dynamic Skeleton interface 12-5, 17-37
  - mapped to C++ 20-99
  - mapping to C 19-40
  - overview of 2-5, 2-10, 6-1
- dynamic\_cast<T\*> 20-61
- DynAny
  - iterating through components of 7-9
  - management overview 7-2
- DynAny API 7-3
- DynAny object
  - basic data type values 7-9
  - copying 7-8
  - creating 7-5
  - destroying 7-8
  - generating an any value from 7-8
  - initializing from an any value 7-8
  - initializing from another DynAny object 7-7
  - interface 7-7

- TypeCode associated with 7-7
- DynAny objects
  - locality and usage constraints 7-5
- DynArray objects
  - interface 7-13
- DynEnum objects
  - interface 7-10
- DynFixed objects
  - interface 7-10
- DynSequence objects
  - interface 7-13
- DynStruct objects
  - interface 7-11
- DynUnion objects
  - interface 7-12

## E

- encapsulation 13-12
  - defined 13-5
- enum 13-11
- enumerated types 3-27
- enumeration type 20-16
- Environment 23-43
- Environment interface
  - OMG PIDL for 20-71
- environment specific inter-ORB protocol for OSF's DCE environment
  - see DCE ESIOP
- environment-specific inter\_ORB protocol
  - see ESIOP
- ESIOP 10-1, 10-4
- Example of how to handle the CORBA-Exception parameter 22-27
- exception 1-7
- ExceptionDef interface
  - OMG IDL for 8-26
- Exceptions 23-4, 23-30
- exceptions 20-62
  - COM and CORBA compared 16-12
  - COM exception structure example 16-17
  - mapped to COM error codes 16-45, 17-34
  - mapped to COM interfaces 16-20
  - mapped to programming languages 19-4
- Exceptions, Application-Specific 23-32
- Exceptions, Example 23-33
- Exceptions, Identifier 23-31
- Exceptions, Members 23-31
- Exceptions, Standard 23-31
- expression
  - context 3-33
  - raises 3-32
- Extensions to COBOL 85 22-49

## F

- federation 11-6
- fixed-length 19-11
- float type 20-15, 21-11
- floating point data type 13-8
- floating point type 3-24
- fooHelper 24-2
- fooHolder 24-2
- foreign object system

- integration of 2-18
- Forward Declaration 23-17
- Forward Declarations 23-3, 23-13
- full bridge 12-2
- fully scoped names
  - defined 3-35

## G

- general inter-ORB protocol
  - see GIOP
- generated class 20-6
- generic pointer 20-56
- get function 20-61
- get\_interface operation 4-5
  - OMG IDL for 4-5
- get\_interface() operation 8-8
- GIOP 10-3, 11-21
  - alignment for primitive data types 13-6
  - and language mapping 13-10
  - and primitive data types 13-3, 13-5, 13-10
  - any type 13-17
  - array type 13-11
  - cancel request header, OMG IDL for 13-26
  - close connection message 13-29, 13-31
  - constructed data types 13-10
  - context pseudo object 13-18
  - exception 13-18
  - floating point data type 13-8
  - goals of 13-2
  - implementation on various transport protocols 13-30
  - integer data types 13-7
  - locate reply header, OMG IDL for 13-28
  - locate request header, OMG IDL for 13-27
  - mapping to TCP/IP transport protocol 13-33
  - message header, OMG IDL for 13-20
  - message type 13-19
  - primitive data types 13-6
  - principal pseudo object 13-18
  - relationship to IIOP 10-3
  - reply message, OMG IDL for 13-24
  - RepositoryId parameters 13-16
  - request header, OMG IDL for 13-22
  - TCKind 13-13
  - typecode 13-13
- GIOP module 13-22, 13-27, 13-28
  - OMG IDL for 13-37
- global name 3-36, 19-5
  - and inheritance 3-36
  - and Interface Repository ScopedName 8-10
  - and Smalltalk 21-5
- Global Names 23-11

## H

- hash operation 4-6
- hexadecimal string 11-20
- HRESULT 16-11, 17-5, 17-10, 17-36
  - constants and their values 16-12

## I

- IConnectionPointContainer interface 17-52
- ICORBA\_Context interface 16-31

- ICORBAFactory interface 15-24, 15-37
- ICORBAObject interface 15-27
- ICustomer
  - Get\_Profile interface 16-25
- identifier 3-15
- Identifiers 23-6
- IDispatch interface 15-4, 15-11, 17-10
- IDL file 23-11
- IDLType interface 8-9
- IEnumConnectionPoints interface 17-54
- IEnumConnections interface 17-54
- IForeignException interface 17-30
- IForeignObject interface 15-26, 15-36, 17-17
- IID 15-17, 15-29, 16-44
- IIOP 11-16, 11-21, 13-2, 13-33, 15-17, 15-32, 15-33
  - defined 13-33
  - host 13-36
  - object key 13-36
  - port 13-36
  - relationship to GIOP 10-3
  - version 13-35
- IIOP module 11-18, 13-34
  - OMG IDL for 13-39
- IIOP profile
  - OMG IDL for 13-34
- IMonikerProvider interface 15-23, 15-36
- implementation
  - defined 1-9, 2
  - model for 1-8
- Implementation Repository
  - overview of 2-11
- implementation skeleton
  - overview of 2-9
- implicit context 11-10, 12-7
- include 23-11
- infix operator 3-20
- Inheritance 23-4, 23-13
- inheritance
  - COM mapping for 16-25
  - OLE Automation mapping for 17-5
- Inheritance and Interface Names 22-6
- inheritance, multiple 15-11
- inheritance, single 17-5
- Initialization interfaces 17-39
- in-line bridging 12-2
- integer data type 13-7
- integer tdata type 3-24
- interface 1-5
  - defined 1-6, 2
- interface identifier
  - see IID 15-17
- interface inheritance 20-7
- interface object 8-7
- interface package 23-12
- Interface Repository 2-5, 13-16
  - AliasDef, OMG IDL 8-22
  - and COM EX repository id 17-31
  - and COM mapping 15-11
  - and identifiers 8-9
  - and request level bridging 12-6

- ArrayDef, OMG IDL 8-25
- AttributeDef, OMG IDL 8-26
- Contained interface, OMG IDL 8-11
- Container 8-9
- Container interface, OMG IDL 8-13
- ExceptionDef interface 8-26
- IDLType 8-9
- inserting information 8-4
- InterfaceDef, OMG IDL 8-29
- IObject interface 8-9
- IObject interface, OMG IDL 8-10
- legal typecodes 8-40
- location of interfaces in 8-8
- mapped to OLE type library 16-51
- ModuleDef interface, OMG IDL 8-19
- OMG IDL for 8-44
- OperationDef, OMG IDL 8-27
- overview of 2-11, 8-2
- PrimitiveDef, OMG IDL 8-23
- Repository interface, OMG IDL 8-17
- SequenceDef, OMG IDL 8-24
- StringDef, OMG IDL 8-23
- StructDef, OMG IDL 8-20
- TypeCode 8-42
- TypeCode interface, OMG IDL 8-36
- InterfaceDef 8-8
  - OMG IDL for 8-29
- InterfaceDef interface 16-51
- Interfaces 23-2, 23-12, 23-44
- Interfaces, Server Side 23-44
- Internet inter-ORB protocol
  - see IIOP
- interoperability
  - architecture of 11-1
  - compliance 10-5
  - domain 11-5
  - examples of 10-5
  - object service-specific information, passing 11-21, 13-4
  - overview of 10-2
  - primitive data types 13-6
  - RFP for 11-1
- interoperability, compliance xxvi
- interoperable object reference
  - see IOR
- interworking 15-13
  - any type 16-38
  - array to collection mapping 17-49
  - Automation View Dual interface 15-31
  - Automation View interface 15-30, 15-32
  - BindingIterator interface, mapped to ODL 17-60
  - bridges 15-33
  - COM aggregation mechanism 17-37
  - COM data types mapped to CORBA types 16-2
  - COM Service 17-51
  - COM View interface 15-29, 15-30
  - compliance xxvi
  - ConnectionPoint Service 17-52
  - CORBA\_Context interface 16-31
  - CORBAComposite interface 16-50
  - CosNaming interface
    - mapped to ODL 17-56
  - DCORBATypeCode interface 17-24
  - DCORBAUnion interface 17-22
  - DCORBAUserException interface 17-31
  - DICORBAAny interface 15-27, 17-24
  - DICORBAFactory interface 15-24, 17-26, 17-27
  - DICORBAStruct interface 17-21
  - DICORBASystemException interface 17-33
  - DICORBAUnion interface 17-22
  - DICORBAUserException interface 17-31
  - DIForeignComplexType interface 17-20
  - Dual interface 15-12, 17-4
  - HRESULT 16-11, 17-5, 17-10, 17-36
  - IConnectionPointContainer interface 17-52
  - ICORBAFactory interface 15-24, 15-37
  - ICORBAObject interface 15-27
  - ICustomer
    - Get\_Profile interface 16-25
  - IDispatch interface 15-4
  - IDispatch interface 17-10
  - IEnumConnectionPoints interface 17-54
  - IEnumConnections interface 17-54
  - IForeignException interface 17-30
  - IForeignObject interface 15-26, 15-36, 17-17
  - IMonikerProvider interface 15-23, 15-36
  - inheritance, mapping for 16-49
  - IORBObject interface 15-28
  - IProvideClassInfo interface 16-32, 16-51
  - ISO Latin1 alphabetic ordering model 17-8
  - ISupportErrorInfo interface 16-15
  - ITypeFactory interface 17-29
  - ITypeInfo interface 16-32, 16-51
  - IUnknown interface 17-10
  - mapping between OMG IDL and OLE, overview 17-3
  - MIDL and ODL data types mapped to CORBA types 16-32
  - MIDL data types 16-2
  - MIDL pointers 16-43
  - multiple inheritance 17-6
  - OLE data types 17-9
  - OLE data types mapped to CORBA types 17-42
  - pseudo object mapping 16-28
  - QueryInterface 15-11, 17-7
  - sequence to collection mapping 17-49
  - SetErrorInfo interface 16-15
  - SimpleFactory interface 15-23
  - single inheritance 17-5
  - target 15-6
  - types of mappings 15-8
  - VARIANT 16-40, 17-5, 17-48
  - VARIANT data types 16-40
  - view 15-5
  - View interface program id 15-31
- interworking object model 15-3
- IOP module
  - and DCE ESIOP 11-21
  - and GIOP 11-21
  - and IIOP 11-21
  - OMG IDL for 11-15
- IOR 11-15, 11-18, 11-19, 13-28, 14-5
  - converting to object reference 11-20



- externalized 11-20
- IORBObject interface 15-28
- IProvideClassInfo interface 16-32, 16-51
- IROBject interface 8-9
  - OMG IDL for 8-10
- is 20-8
- is\_equivalent operation 4-7
- is\_nil operation 20-8
- ISupportErrorInfo interface 16-15
- ITypeFactory interface 17-29
- ITypeInfo interface 16-32, 16-51
- IUnknown interface 17-10

## L

- language mapping
  - overview 2-8
- left-shift-assign operator 20-47
- Literals 23-6
  - Literals, Character 23-7
  - Literals, Floating-Point 23-6
  - Literals, Integer 23-6
  - Literals, String 23-8
- little endian 13-7
- logical\_type\_id string 4-6
- long double type 21-11
- long long type 21-11
- long type 20-15, 21-11

## M

- magic 13-20, 13-37, 13-38
- Mapping for Interfaces 22-5
- Mapping IDL Identifiers to a COBOL Literal 22-3
- Mapping IDL Identifiers to a COBOL Name 22-2
- mapping IDL to Java
  - basic types 24-3
  - helper classes 24-9
  - Java ORB portability interfaces 24-49
  - mapping for array 24-18
  - mapping for certain nested types 24-29
  - mapping for constant 24-10
  - mapping for enum 24-11
  - mapping for exception 24-22
  - mapping for interface 24-19
  - mapping for sequence 24-17
  - mapping for struct 24-13
  - mapping for the Any type 24-26
  - mapping for typedef 24-30
  - mapping for union 24-14
  - mapping psuedo objects 24-31
  - modules 24-3
  - names 24-2
  - server-side mapping 24-48
- Mapping of BOA's Dynamic Implementation Routine to COBOL 22-41
- Mapping of the ServerRequest to COBOL 22-40
- Mapping Pseudo Objects to COBOL 22-29
- mediated bridging 11-8
- Memory Management 23-5
- method 1-8
- Microsoft Interface Definition Language
  - see MIDL 15-4

- MIDL 15-4
  - transformation rules 15-13
- modifier function 20-33
- ModuleDef interface
  - OMG IDL for 8-19
- Modules 23-12
- multiple inheritance 3-17, 15-11, 17-6
- MultipleComponentProfile 11-16

## N

- NamedValue 23-37
- NamedValue interface
  - OMG PIDL for 20-73
- NamedValue type 5-2
- Names 23-5, 23-10
- namespace 20-3, 20-116, 21-6
- NamingContext 12-7
- NamingContext interface
  - mapped to Smalltalk 21-18
- Narrowing 23-15
- nested scope
  - and definitions 3-35
- Nil 23-16
- nil 21-8
- nil object reference 20-10
- null pointer 20-51, 20-64
- NVList 16-28, 21-23, 23-37
- NVList interface
  - add\_item operation 5-12
  - create\_list operation 5-12
  - create\_operation\_list 5-13
  - free operation 5-12
  - get\_count operation 5-13
  - OMG PIDL for 20-74
- NVList operation
  - free\_memory operation 5-13
- NVList type 5-2, 20-100

## O

- Object 23-16, 23-42
- object
  - context 5-13
  - CORBA and COM compared 15-9
  - defined 3
  - implementation 1-9, 2-7
  - invocation 2-9, 2-10
  - mapping to programming languages 19-3
  - reference 2-8
  - reference canonicalization 11-13
  - reference embedding 11-12
  - reference encapsulation 11-13
  - references, stringified 11-19
  - request 11-3
- object adapter 2-6, 2-9, 2-14
  - and request level bridging 12-6
  - functions of 2-15
  - overview of 2-5, 2-10
- object class 20-8
- Object Definition Language 15-4
- object duplicate operation
  - OMG PIDL for 4-5

- object identifiers
    - and hash operation 4-6
  - Object interface
    - create\_request operation 4-4
    - OMG PIDL for 4-4, 20-87
  - object key 13-18
  - Object Management Group xxiii
    - address of xxv
  - Object model 1-2
  - Object Reference 23-13
  - object reference 20-6
    - and COM interface pointers 15-4
    - obtaining for View interface 17-40
    - testing for equivalence 4-7
    - union members 20-34
  - Object Reference Operations 23-14
  - object reference variable type 20-7
  - Object References 22-5
  - object references
    - obtaining for automation controller environments 17-26
  - Object References as Arguments 22-5
  - Object Request Broker xxiv
    - explained 2-2
    - how implemented 2-6
    - interfaces to 2-2
    - sample implementations 2-11, ??-2-13
  - Object Services xxiv
    - and GIOP module 13-22
    - and interoperability 12-7
    - and IOP module 11-20
    - Life Cycle 15-20, 15-22, 15-23, 16-50, 17-26
    - Naming 12-7, 15-25, 17-26, 17-27, 17-39
    - Naming, sample mapping to OLE 17-51, 17-55
    - Relationship 10-5
    - tags for A-1
    - Transaction 11-10
  - object type 1-5
  - Object\_ptr 20-8
  - object\_to\_string operation 4-3
    - OMG PIDL for 4-3
  - Object\_var 20-8
  - Objects 1-3
  - octet type 3-25, 13-4, 13-10, 20-15, 20-16, 21-12
  - ODL 16-4, 17-1
  - OLE Automation 15-4
    - basic data types 17-9
    - basic data types mapped to CORBA types 17-42
    - relationship to OMG IDL 17-3
    - transformation rules 15-13
  - OLE automation controller 17-2
  - OMG IDL
    - overview of 2-8
    - relationship to OLE 17-3
    - syntax of 3-14
  - OMG IDL global name 3-36
  - OMG IDL struct
    - mapping to C++ 20-27
  - OMG IDL tags
    - requests to allocate 11-19, A-1
  - OMG IDL-to-programming language mapping
    - overview 2-8
  - oneway 16-23, 20-61, 20-62, 3
  - opaque data type 13-5
  - operation 20-7
    - attribute, syntax of 3-31
    - declaration, syntax of 3-31
    - defined 1-6
    - mapping to programming languages 19-4
    - signature of 1-6
  - OperationDef
    - OMG IDL for 8-27
  - Operations 17-33, 23-3, 23-35, 23-44
  - Operations, Server Side 23-44
  - operator 20-47
  - operator< 20-47
  - operator> 20-23
  - operator>=> 20-50
  - operator[] 20-38
  - Operators 23-9
  - ORB 23-42
    - backbone 11-11
    - connecting 8-4
    - core 11-3
    - kernel 11-3
  - ORB initialization
    - mapped to C 19-44
  - ORB Interface
    - overview of 2-10
  - ORB interface
    - and create\_list operation 5-12
    - and create\_operation\_list operation 5-13
    - and NVList objects 5-11
    - mapping to programming languages 19-5
    - OMG PIDL for 20-83
  - ORB Services 11-3, 11-7
    - how selected 11-4
    - vs. Object Services 11-3
  - ORB Supplied Functions for Mapping 22-46
  - ORB\_init operation 19-45, 20-86
    - mapped to C++ 19-45, 20-86
- P**
- parameter
    - defined 1-7
  - parameter declaration
    - syntax of 3-32
  - POA Interface 9-30
    - locality constraints 9-30
  - pointer type 20-7
  - Portable Object Adaptor
    - abstract model description 9-2
    - AdaptorActivator interface 9-19
    - creating 9-30, 9-48
    - creating object references 9-7
    - creation 9-6
    - destroying 9-31
    - dynamic skeleton interface 9-12
    - finding 9-31
    - implicit activation 9-10
    - Implicit Activation policy 9-29
    - interface 9-30
    - model architecture 9-4



- model components 9-2
- multi-threading 9-11
- overview 9-1
- request processing 9-9
- root POA 9-48
- ServantActivator interface 9-21
- ServantLocator Interface 9-24
- ServantManager interface 9-20
- SYSTEM\_ID policy 9-48
- usage scenarios 9-47
- Portable Object Adaptor
  - policy objects 9-25
- PortableServer
  - UML description of 9-46
- PowerPoint.Slide.7 9-5
- pragma directive
  - and Interface Repository 8-32
  - id 8-32
  - prefix 8-32
  - use in Smalltalk mapping 21-14
- PrimitiveDef
  - OMG IDL for 8-23
- Principal 23-40
- principal 13-12, 13-23
- principal pseudo object 16-28, 16-31
- profile
  - tags for A-1
- property name 5-14
- pseudo keyword 20-69

**Q**

- qualified name 3-35
- QueryInterface 15-11, 17-7

**R**

- readonly 20-61
- reference encapsulation 12-5
- reference model xxiv
- reference translation 12-5
- Relationship Service 10-5
- release operation 4-5, 20-8
- release parameter 20-37
- replace function 20-56
- Repository interface
  - OMG IDL for 8-17
- RepositoryId
  - and COM interface identifiers 16-44
  - and COM mapping 16-11
  - and pragma directive 8-32
  - format of 8-31
- Request 23-38
- Request interface
  - add\_arg operation 5-7
  - delete operation 5-8
  - get\_next\_response operation 5-10
  - get\_response operation 5-10
  - invoke operation 5-8
  - OMG PIDL for 20-77
  - send operation 5-8
  - send\_multiple\_requests operation 5-9
- request level bridging 12-2

- types of 12-6
- Requests 1-3
- result
  - defined 1-7
- right-shift-operator 20-50
- RPC 14-20, 14-24
- RTTI 20-61
- Run time type information
  - see RTTI

## S

- SAFEARRAY 15-10, 16-40, 17-19
- scoped name identifier 3-35
- Scoped Names 22-3
- scoped\_name 3-17
- scoping
  - and C language mapping 19-6
  - and C++ mapping 20-4
  - and identifiers 3-35
  - explained 3-35
- see ODL 15-4
- Sequence 23-23
- sequence octet 13-13, 13-18
- sequence type 3-25, 3-27, 3-32, 13-11, 20-35, 21-15
- Sequence Types 23-23
- SequenceDef
  - OMG IDL for 8-24
- server 20-88, 23-43, 4
- ServerRequest
  - mapped to C 19-40
  - mapped to C++ 20-99
- ServerRequest pseudo interface
  - mapped to C 19-40
- Service Type Repository Module 23-45, 23-50, 23-51, 23-52, 23-53, 23-54, 23-55
- ServiceContext 11-21
- ServiceID 11-22
- set function 20-61
- SetErrorInfo interface 16-15
- Short 20-16
- short type 20-15, 21-10
- signature 4
- SimpleFactory interface 15-23
- single 20-64
- sizeof(T) 20-5
- skeleton class 20-91, 20-93
- slice 20-42, 20-65
- Smalltalk 21-11
  - aBindingStruct 21-13
  - aBool 21-9
  - aCORBAObject 21-19, 21-22
  - active Process message 21-16
  - add\_arg operation 21-19
  - addArg instance method 21-19
  - aDiscriminator instance method 21-15
  - any 21-12
  - argList 21-9
  - array class 21-15
  - array type 21-15
  - Association 21-20
  - at message 21-12, 21-17

- boolean 21-12
  - char 21-12
  - Character 21-12
  - Common Base 21-5
  - CORBAConstants 21-10, 21-12, 21-17
  - corbaContext message 21-16
  - CORBAContext protocol 21-20, 21-22
  - CORBAEnum protocol 21-12
  - CORBAExceptionEvent protocol 21-16
  - CORBAExceptionValue protocol 21-17, 21-19
  - CORBANamedValue protocol 21-22, 21-23
  - CORBAObject protocol 21-21
  - CORBAORB protocol 21-21
  - CORBAParameter protocol 21-8
  - corbaRaise message 21-17
  - CORBARequest protocol 21-19
  - CORBAUnion protocol 21-15
  - create\_child operation 21-20
  - create\_operation\_list operation 21-22
  - create\_request operation 21-21
  - createChild instance method 21-20
  - createOperationList instance method 21-22
  - createRequest 21-9, 21-21
  - cxt 21-19
  - delete instance method 21-20
  - delete operation 21-20
  - design of mapping 21-4
  - Dictionary 21-18
  - Dictionary class 21-13, 21-15
  - discriminator instance method 21-15
  - duplicate 21-21
  - exceptions 21-15
  - explicit vs implicit mappings 21-14
  - flags instance method 21-22
  - float 21-11
  - garbage collection 21-8, 21-17
  - get\_next\_response operation 21-22
  - get\_response operation 21-20
  - getDefaultContext instance method 21-22
  - invoke instance method 21-19
  - invoke operation 21-19
  - invokeOneway instance method 21-19
  - long 21-11
  - long double 21-11
  - long long 21-11
  - memory management 21-8, 21-21
  - memory management for object references 21-17
  - mini-glossary 21-24
  - name instance method 21-22
  - namespace 21-6
  - nil 21-8
  - NVlist type and OrderedCollection class 21-23
  - object\_to\_string operation 21-21
  - objectToString instance method 21-21
  - octet 21-12
  - operation 21-9
  - OrderedCollection class 21-15
  - overview of mapping 21-3
  - pollNextResponse instance method 21-22
  - pollResponse instance method 21-20
  - Processor variable 21-16
  - reference books 21-5
  - release operation 21-21
  - reqFlags 21-9
  - request 21-9
  - restOfName 21-19
  - result 21-9
  - send instance method 21-20
  - send operation 21-20
  - send\_multiple\_requests operation 21-22
  - sendMultipleRequests instance method 21-22
  - sequence 21-15
  - set and get value instance methods 21-20
  - set and get value operations 21-20
  - set value operations 21-20
  - short 21-10
  - String class 21-15
  - string type 21-15
  - struct type 21-13
  - underscore characters in mapping 21-7
  - unsigned long 21-11
  - unsigned long long 21-11
  - unsigned short and long 21-11
  - value instance method 21-15, 21-22
  - Value instance methods 21-20
  - wchar 21-12
  - split allocation
    - avoiding errors with 20-63
  - statically-initialized 20-41
  - string 23-20
  - string type 3-28, 3-32, 13-11, 20-17
  - String Types 23-26
  - string union members 20-34
  - string\_to\_object operation 4-3
    - OMG PIDL for 4-3
  - String\_var 20-17
  - StringDef
    - OMG IDL for 8-23
  - struct type 3-25, 13-11
  - StructDef
    - OMG IDL for 8-20
  - stub 4
  - stub interface 2-8, 2-9
  - subject 3-34
  - Subsystem 23-11
  - Summary of IDL Constructs to Ada Constructs 23-2
  - SystemException 20-59
- T**
- T \*data constructor 20-37
  - T\_ptr\* 20-38
  - T\_var 20-21, 20-118
  - tag
    - component 11-19
    - protocol 11-19
    - requests to allocate A-1
  - TAG\_MULTIPLE\_COMPONENTS tag 11-16, 11-19
  - Tagged Types 23-2
  - target 15-6, 15-34
  - Tasking 23-5, 23-36
  - TCKind 13-13
  - TCP/IP 13-30, 13-33

- template 20-3, 20-93
- test 19-3
- this pointer 20-98
- throw exception 20-60, 20-97
- tie class 20-94
- top 20-13
- Transaction Service 11-10
- transfer syntax
  - between ORBs and inter-ORB bridges 13-3
- transparency 11-4
- transparency of location 11-2
- type 20-49
- type function 20-57
- type specifier
  - syntax of 3-22
- type unknown to the receiver 20-46
- TypeCode 5-3, 16-28, 20-46, 23-28, 23-40
  - OMG IDL for 8-42
- typecode 21-4
- TypeCode interface
  - OMG IDL for 8-36
- TypeCode\_ptr 20-57, 20-81
- typedef 20-44
- Typedefs 23-28
- Types 23-4
- types
  - any 1-4
  - basic 1-4
  - constructed 1-5
  - defined 1-4
  - interface 1-5
  - legal values 1-5
- Types, Any 23-29
- Types, Arithmetic 23-21
- Types, Array 23-26
- Types, Boolean 23-21
- Types, Enumeration 23-22
- Types, Exception 23-30

- Types, Sequence 23-23
- Types, Size Requirements 23-20
- Types, String 23-26
- Types, Structure 23-22
- Types, TypeCodes 23-28
- Types, Typedefs 23-28
- Types, Union 23-23
- type-safe 20-46

## U

- unbounded sequence 19-11, 20-21
- unbounded string 19-11, 20-21
- Unicode 15-10, 16-36, 17-12
- union member 20-31
- union type 3-26, 13-11, 21-4
- unsigned long long type 21-11
- unsigned long type 20-15, 21-11
- unsigned short type 20-15, 21-11
- UserException 20-58

## V

- value function 20-57
- value instance method 21-15
- variable-length 19-11
- VARIANT 16-40, 17-5, 17-30, 17-48
  - OLE data types 16-40
- view 15-5, 15-21
- View interface 15-31
- Visual Basic 15-9
- void\* 20-57

## W

- wchar type 21-12
- Widening 23-15
- Windows System Registry 15-24, 17-2, 17-25

## X

- X/Open xxiv

