Source: orig/./abl2537-1.doc.pdf
PagePixelOvelayIndex[%]: 58.7 : FeatureDistanceError[mm]: 26.2 : HorizLinePositionError[mm]: 0.68 : TextHeightError[mm]: 19.60 : LineNumDifference: -3

Datapool replicator conceptional specification

1 Table of contents

1 Table of contents 1
2 The differences between OPC and our replicator 1
3 The basic concepts 1
3.1 The server concept ..... **Error! Bookmark not defined.**
3.2 The client concept ..... **Error! Bookmark not defined.**
7.1 List of objects ..... **Error! Bookmark not defined.**
7.2 The common properties and methods ..... **Error! Bookmark not defined.**
8 Demo programs ..... **Error! Bookmark not defined.**
8.1 The simplest server ..... **Error! Bookmark not defined.**
8.2 The simplest client in C++ ..... **Error! Bookmark not defined.**
8.3 The simplest client in Visual Basic ..... **Error! Bookmark not defined.**

2 The differences between OPC and our replicator

It looks like that our replicator duplicates functionality of the OPC. But it is not the case. Replicator optimizes amount of communicated data and thus it is suitable for connection through dial up phone lines with low capacity.

Datapool replicator
+ is very simple
+ could be used for fast time trending
+ allow transmissions of any user type of data
+ minimizes amount of data through communication line
+ allow some special features like rejecting points, functional and duplicative points.
- do not support hierarchical data
- objects are responsible to pack itself into binary stream with their own C++ code
- is not industrially standardized

OPC
+ it is industrial standard
+ allow grouping data into complex data structures
- communication has big overhead
- no user type support
- implementation of OPC server is quite complex

3 The basic concepts

The main idea is that we want to use one object from one or more computers and we want to synchronize data automatically.

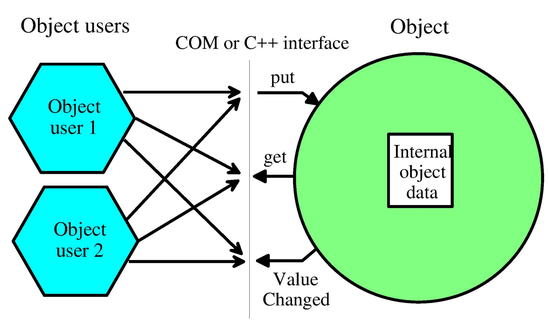


Figure 1 Communication with single object

The normal usage is shown on the figure. But the object data could be accessed from one process only. Therefore the next solution is proposed.

Target: L05M_CLAYOUTLIN/./abl2537-1.doc.L05M_CLAYOUTLIN.pdf

Datapool replicator conceptional specification

1 Table of contents

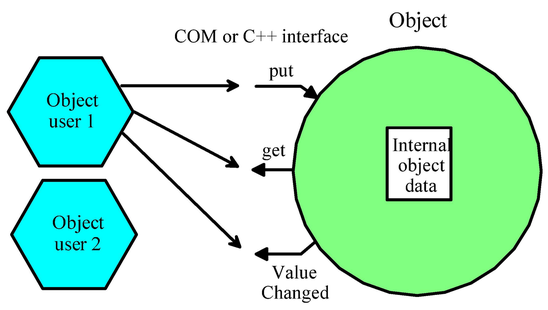
1 Table of contents 1
2 The differences between OPC and our replicator 2
3 The basic concepts 2
3.1 The server concept 4
3.2 The client concept 4
7.1 List of objects 18
7.2 The common properties and methods 18
8 Demo programs 19
8.1 The simplest server 19
8.2 The simplest client in C++ 20
8.3 The simplest client in Visual Basic 21

2 The differences between OPC and our replicator

It looks like that our replicator duplicates functionality of the OPC. But it is not the case. Replicator optimizes amount of communicated data and thus it is suitable for connection through dial up phone lines with low capacity.

Datapool replicator
+ is very simple
+ could be used for fast time trending
+ allow transmissions of any user type of data
+ minimizes amount of data through communication line
+ allow some special features like rejecting points, functional and duplicative points.
- do not support hierarchical data
- objects are responsible to pack itself into binary stream with their own C++ code
- is not industrially standardized

OPC
+ it is industrial standard
+ allow grouping data into complex data structures
- communication has big overhead
- no user type support
- implementation of OPC server is quite complex

3 The basic concepts

The main idea is that we want to use one object from one or more computers and we want to synchronize data automatically.