

# Transparent gateway

**D2000 GateWay Client** may be configured also for direct transfer of values of selected objects from the source D2000 System (shows data through the **D2000 GateWay Server**) to the destination one without the need to configure objects of "Remote tag" type.

This transparent gateway enables transferring values of these types of D2000 objects:

- [Processes](#)
- [Communication lines](#)
- [Stations](#)
- [I/O tags](#)
- [User variables](#)
- [Eval tags](#)
- [Structured variables](#)

The value transfer is executed only on the basis of verification of object match between D2000 Systems. To transfer the values, select the particular objects or group of objects filtered by mask and then enter them into the [start parameters](#) of the *GateWay Client* process.

The destination system tries to verify whether the required objects exist in the source system and if so, it ensures transferring the value of these objects. If the source system does not contain a particular object, it will have an invalid value in the destination system. If the list of objects is changed (any object is added or deleted) while running the *GateWay Client* in the destination system, which corresponds with the mask, the proper action is executed towards the source system (logging on/off of object).

No values are transferred from the destination system to the source system. The value transfer is always **unidirectional**, so the transparent gateway works as a **Data Diode**.

The structured variables can be transferred only as a whole.

The feature [Destination column](#) for I/O tags is not supported. A user must select the values of the given destination structure that are to be transferred.

The *GateWay Client* process can be operated at the same time in the standard [remote tags](#) mode and as the transparent gateway.

## Use of transparent gateway

The historical mode of the gateway is a solution in cases when:

- changes or modifications to the functional and deployed system are made and these are first tested on a parallel system,
- testing of new HW is carried out during the planned upgrade of the existing system,
- other testing or debugging purposes.

## Limitations

You should realize that the transparent gateway assigns the values to objects that are children of some D2000 processes. Therefore, these processes must be **stopped** in the destination D2000 System. These are usually the [communication processes](#) and [calculation processes](#).

The user cannot change the value on the destination D2000 System (controlling values). As the parent communication process does not run, the setting of value ends with an error.

## Start parameters

To activate the transparent gateway, start the *GateWay Client* with one or more **/ROV** parameters (Read Object Values).

Syntax: `/ROVobj_mask1[+][.][obj_mask2][+][.]......`

**/ROV** parameter is followed by a name (or more names separated by a comma) of an object or mask for more objects. To transfer the object values together with the child, add the '+' (plus) character at the end.

Examples:

/ROVSELF.KOM+	Ensures the value transfer of the SELF.KOM process together with its children (lines, stations, I/O tags).
/ROVM.*	It is used for value transfer of the I/O tags whose name starts according to the ordinary naming 'M.'.
/ROVM.OBJ1,M.OBJ2	Value transfer of objects M.OBJ1 and M.OBJ2.

## Object exclusion

If it is required not to transfer the values of specific objects, it is possible to start the *GateWay Client* process with one or more parameters **/XOV** (eXclude Object Values).

Syntax: **/XOVobj\_mask**

Objects whose names match the specified mask will not be transferred (not even their descendants). The **/XOV** parameter has a higher priority than the **/ROV** parameter.

Example: The transfer of all structured variables except system structures can be achieved by specifying **/ROVSV.\* /XOVSV.\_System\***

### Changing object names

If it is necessary to change object names (i.e. on the *GateWay Client* side, D2000 objects are named differently than on the *GateWay Server* side), it is possible to start the *GateWay Client* process with one or more **/MON** (Modify Object Names) parameters.

Syntax: **/MON<expression,replacement>**

Example: Deleting the string "MPPS." in the names of the opened objects we achieve by specifying **/MONMPPS\.**

Example: Replacement MPPS -> MES is achieved with the parameter **/MONMPPS,MES**

Note: at most one substitution is performed on the D2000 object name (expression is searched for from the beginning of the name). For example, if an I/O tag M.MPPS.test.MPPS exists, then the parameter **/MONMPPS,MES** causes the I/O tag M.MES.test.MPPS to be opened on the remote side.

## Pairing objects by UID

A transparent gateway supports pairing not only by object names but also by UID objects. This feature is enabled by the start parameter **/UID**, which must be used when starting both the *GateWay Client* and *GateWay Server* processes.

This feature can be used to implement the functionality of a transparent gateway regardless of changes in the names of the objects (provided that they have the same UID), i.e. objects can be renamed.

### Filtering of process alarms

When transferring the values to the destination system, the values of process alarms are usually filtered. To transfer values together with the process alarms, start *GateWay Client* with the **/ROA** parameter.

### Invalidation of values of non-found objects

If the transparent gateway does not find the desired object on the side of the gateway server, it invalidates the value of the object on the side of the gateway client. This feature can be disabled by the **/NOINV** start parameter.

### Optimization for XML import

When performing XML import of multiple objects to the destination system, the D2000 Server used a lot of CPU for performing refreshes of object lists. Refreshing can be delayed by starting the *GateWay Client* with the **/GTWDL<seconds>** start parameter (refresh will be performed with a delay equal to the specified **seconds** parameter - the recommended value is 10).

## Diagnostic information

To activate the detailed debug information, start the *GateWay Client* with these parameters:

<b>/E+DBG.GTW_TR</b>	Enables listing of the debug information about the objects for operations of the transparent gateway.
<b>/E+DBG.GTW_TR.VAL</b>	Enables listing of the debug information about the value transfer of selected objects to the destination system.

This listing of debug information can be enabled /disabled also while running the *GateWay Client* through the [D2000 System Console](#).

Tell command **TRGTW**

After receiving the **TRGTW** command, the *GateWay Client* lists the status of all required objects.



#### Blog

You can also read about the transparent gateway in the following blogs:

- [Communication in testing environments](#)
- [Connection of six SCADA systems](#)
- [Mirroring objects between D2000 systems](#)



**Related pages:**

[Remote tags](#)