

Incorporating Existing Applications

Applications and objects may be incorporated into PAC without being compiled or recompiled. Incorporation allows you to place under PAC control applications or objects that exist in catalog form only.

The PAC incorporation facility is used for migrations from environments where development or maintenance is performed to Control, or to a test or production status type.

Once an application has been linked to an incorporate status type, the entire application or just selected objects can be incorporated into PAC. PAA treats objects incorporated into a production status type like any other objects.

Saved, cataloged, and cross-reference (Xref) objects can be incorporated in the following combinations:

Saved
Cataloged
Saved/cataloged
Cataloged + Xref
Saved/cataloged + Xref

Xref data is optional, but must be accompanied by the corresponding cataloged object; otherwise, it will be rejected.

An object may be incorporated only if a previous version of the object was not compiled in PAC. PAC does not allow an object previously migrated into PAC from a development or maintenance status type to be incorporated, because incorporation could affect the integrity of not only the object itself, but related objects and even the entire application. Since objects are not compiled when they are incorporated into PAC, PAC has no way of synchronizing and controlling subordinate and other objects that may be used by the incorporated object.

There is no restriction on the number of times an object can be incorporated into PAC; however, PAC always considers the cataloged object as "alien" because it was compiled outside the control of PAC and its used and used-by objects and/or Xref data may be inconsistent.

With incorporated objects, certain interrelationships may be lost; for example, object usage information established from the Xref data derived from the compile.

This chapter covers the following topics:

- Incorporation Options
- Incorporation Paths
- Incorporation Object List
- Authorizing and Submitting an Incorporation Event

Incorporation Options

Objects may be incorporated online or in batch; they may be incorporated from an externally created work file or not:

- If the Include option is specified, objects are by definition incorporated in batch from an externally created work file. The work file must have been created using the Natural NATUNLD utility.

Note:

Do not use a work file created by the Natural Transfer Utility, except when incorporating Natural Open System objects.

- Without a work file as input, objects are incorporated directly. The process is the same as a migration from development, including setting up the object list; however, objects are not compiled.

If the Include option is specified, the INCORPORATE batch sample job is used. Step 1 of this job should assign Work File 1 (CMWKF01) to the external work file. The Include flag must be set in the migration path definition or at the time the event is authorized.

If the incorporated objects are then to be migrated to another status and this migration uses an output work file (the work file option is set), an extra step, the TRNUNLD command, is needed to unload the necessary objects. Refer to Job Steps: Batch With Work File Option for more information.

If only an input and not an output work file is required (the Include option is specified but the work file option is not set), the JCL should be modified appropriately to exclude the TRNUNLD and the migration load step. Refer to Job Steps: Online or Batch Without Work File Option, Using Include Option for more information.

Incorporation Paths

Objects and applications may be incorporated into PAC using previously defined migration paths from an incorporate status type to Control, a test status type, or a production status type.

If the origin and destination statuses have the same library, database and file number in their application status links, PAC treats the migration as a logical one and does not unload the objects. To ensure that the objects to be incorporated are unloaded, use a different Natural library name or a different database or file number for the destination status.

If objects are incorporated from an external work file (the Include option is specified), the library selection is determined by the library name on the application status link. For example, if the destination library is LIB01, then the library on the incorporation application status link should also be LIB01.

Incorporation Object List

Natural views/DDMs that exist in catalog form external to PAC can be incorporated into PAC without being regenerated, but only for the initial setup of Predict information in PAC. Because views/DDMs are considered Natural objects, this is a Natural migration. For information about incorporating views/DDMs, refer to Migrating Predict Objects.

If objects are to be incorporated directly (that is, the Copy/Move option is specified indicating that a work file is not to be used as an input), then the object list for incorporation may be set up as for migrations from development, except that no views/DDMs or rules may be specified.

The EXP (Expand) command and special options of the SEL (Select) command use cross-reference (Xref) data from Predict for incorporations.

If objects are to be incorporated from an external work file (that is, the Include option is specified), then the object list must consist of a single entry only which indicates which objects of which type are to be loaded. For example:

*.P loads all program type objects.

S>.M loads all maps with a name greater than "S".

Authorizing and Submitting an Incorporation Event

When the incorporation event is authorized and submitted:

1. PAC selects all objects identified by the object list entry from the library on the application status link.
2. Objects satisfying the selection criteria are assigned a unique version number and added to the PCF system file. The object may exist in saved and/or cataloged form.
3. Once loaded, PAC bypasses the compile stage and processes the object like any other new object from a development status type.
4. The versioned object is stored on the ACF system file.