

# Subnetworks

The job of NET type enables you to define a subnetwork within a main network. This allows you to build nested networks. The subnetwork must already exist when the definition is created. The same subnetwork can be defined in different jobs of the main network. On activation, each active subnetwork is assigned a unique run number. Subnetworks can in turn be invoked within subnetworks, however a subnetwork cannot invoke itself, because this could cause an infinite recursion.

In the job list of the calling network, you can invoke the job list of the defined subnetwork with the line command Z from the calling job of type NET.

This subsection covers the following topics:

- Link to the Main Network
- Time of Activation of a Subnetwork
- Activation and Execution
- Symbol Table Activation for Subnetworks

## Link to the Main Network

An input condition NET-BEGIN and an output condition NET-END must be defined in the subnetwork, as link to the main network. The condition NET-END-NOTOK can be used to set the status "ended not ok".

For more information, see the section Reserved Condition Names in Concepts and Facilities.

Condition	Description
NET-BEGIN	This is an input condition of the first job of the subnetwork.
NET-END	This is an output condition of the last job of the subnetwork.
NET-END-NOTOK	If this condition is set by the subnetwork, then the job of NET type is treated as <b>ended not ok</b> . This condition must be set in addition to NET-END.

## Time of Activation of a Subnetwork

Subnetworks can be activated at two different times:

- at the time of the activation of the calling network  
The subnetwork is activated together with the calling network (i.e., with the job type NET). It will be available from this point in time with run number and active JCL. This is the default.
- at the start time (submission) of the calling job of the type NET  
The subnetwork is only activated if the calling job of the type NET is really started. This can prevent a subnetwork from even being activated if the calling job is not executed at all later on. Moreover, the effort for job activations and loading the JCL is thus compensated for, and the waiting-time for prerequisites within the subnetwork will be shorter.  
Note, however, that the active subnetwork including active JCLs will only exist from a later time onwards.

The **subnetwork activation mode** can be defined as follows:

- as a global default value (see the section Entire Operations Administration Documentation).
- in the definition of the calling job of the type NET (see Defining a Subnetwork).

## Activation and Execution

A subnetwork is activated together with the main network. The subnetwork is assigned its own run numbers. Reserved symbols pass information to the jobs of the subnetwork about which job invoked the subnetwork. These symbols are:

Symbol	Description
P-C-OWNER	Owner of the invoking network.
P-C-NETWORK	The invoking network.
P-C-JOB	Job of the invoking network.
P-C-RUN	Run number of the invoking network.
P-C-SUFFIX	Suffix value of the job which invoked the subnetwork where the symbol is used.
P-C-SYMBOL-TABLE	Symbol table of the invoking network.

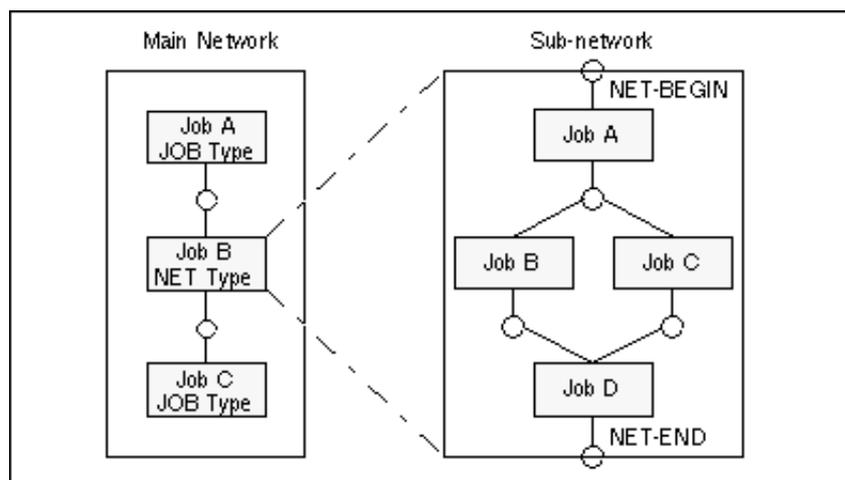
For a detailed description of the symbols, see the section Symbols.

If the subnetwork has a standard symbol table, these values are entered as symbols in the active copy of this table.

This enables you to trace, even through several steps, from where the subnetwork was invoked.

If the job of NET type can be activated (satisfies all input conditions), the condition NET-BEGIN of the subnetwork is set and the subnetwork starts to work: the NET-type job is then **Executing**. It remains in this status until the subnetwork has set the condition NET-END. Only at this point are the end-of-job checking and actions performed which are defined in the invoking network.

If an execution error occurs in the subnetwork, the execution of the invoking network is also blocked. If the subnetwork sets the condition NET-END before its actual termination (or at its very start), the invoking network continues to run and a parallel asynchronous execution of the networks is possible. However, the user him/herself then becomes responsible for later synchronizing the subnetwork with the invoking network.



## Symbol Table Activation for Subnetworks

The order of symbol table activation for subnetworks is:

1. Set caller info into the active table (P-C-... symbols).
2. Activate symbol table(s).
3. Perform symbol modification exit.

The order of symbol table activation for subnetworks is the same as for main networks.