

Sub-networks

The job of NET type enables you to define a sub-network within a main network. This allows you to build nested networks. The sub-network must already exist when the definition is created. The same sub-network can be defined in different jobs of the main network. On activation, each active sub-network is assigned a unique run number. Sub-networks can in turn be invoked within sub-networks, however a sub-network 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 sub-network with the line command Z from the calling job of type NET.

This subsection covers the following topics:

- Time of Activation of a Sub-Network
- Link to the Main Network
- Run

Time of Activation of a Sub-Network

Sub-networks can be activated at two different times:

- at the time of the activation of the calling network
The sub-network 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 sub-network is only activated if the calling job of the type NET is really started. This can prevent a sub-network 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 sub-network will be shorter.
Note, however, that the active sub-network including active JCLs will only exist from a later time onwards.

The **sub-network 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 Sub-network).

Link to the Main Network

An input condition NET-BEGIN and an output condition NET-END must be defined in the sub-network.

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

Run

A sub-network is activated together with the main network. The sub-network is assigned its own run numbers. Reserved symbols pass information to the jobs of the sub-network about which job invoked the sub-network. 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 sub-network 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 sub-network 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 sub-network was invoked.

If the job of NET type can be activated (satisfies all input conditions), the condition NET-BEGIN of the sub-network is set and the sub-network starts to work: the NET-type job is then **Executing**. It remains in this status until the sub-network 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 sub-network, the execution of the invoking network is also blocked. If the sub-network 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 sub-network with the invoking network.

