Environment

Within an SAP system or at least what we're looking at we have two types of programs. We have report. And we have dynpros.

Reports

Reports are just programs that generate lists of data. Or maybe they've got a little bit of interactivity in them. But basically, they supply data to the front-end databases and so on. When a user runs a report, they typically get a selection screen. Once they enter their selection parameters and execute the report, they normally cannot intervene in the execution of the program. The program runs and then displays the output.

Dynpro

Dynpro's are a little bit different. Dynpro's are dynamic programs. They allow the user to intervene in the execution of the program and they work by processing a series of screens that we call dialogue screens.

The user determines the flow of the program itself by choosing which buttons or fields to interact with on the screen. That action then triggers different functions that have been coded within the flow logic of a program.

While we're creating a report, we will be generating interfaces that clusters dynpros for all our selection criteria.

Now don't fret, because most of the work that ABAP people do is actually inside report programs. And even though these programs are classed as reports, they don't always generate output. The report programs are there to process all the logic, reading and writing to the database.

Try to make your systems work let's have a look at word processors. Every program that runs SAP system runs on what we call word processors, and these actually run on the application server. What processes themselves work independently of the computer's operating system and the database that it interacts with? This gives us that program independence that we discussed earlier when looking at the technical architecture of an SAP system.

When an SAP system is initially set up your basis consultants. They're the guys who actually install SAP and keep it running, you know, manage all the memory and so on. They configure SAP in such away that it automatically sets the number of work processes your programs use when they start, and that's the equivalent of like, setting up a predefined number of channels or connections to the database system itself. Each of which tend to have their own different properties and different functions.

You might come across something referred to as the dispatcher. Your SAP system has no technical limits on the number of users who can actually log on and use an SAP system. Generally, the number of users that can access an SAP system is a lot larger than the number of available work processes that your system is configured for.

This is because not everybody is you know, sending instructions to the application server. At exactly the same time. It's a bit like the old mainframe time sharing scenario. And because of this. Users cannot be assigned, you know, a certain number of processes while they're logged on.

It is the dispatcher that controls the distribution of the work processes to the system users. The dispatcher will keep an eye on how many work processes are available and when a user triggers a transaction, the dispatcher's job is to say, here you go. Here's a free work process. Go and use it the dispatcher tries as much as possible to optimize things so that the same work process receives the sequential dialogue steps of an application.

If you can't do this, let’s say because the user takes a long time between click in on different aspects of this screen. He will then, must select a different web process take carry on with the processing of the dialogue program. Is the work process that executes an application, and it's the work process that has access to the memory areas that contain all the data and objects that an application uses. It also makes three very important elements available.

The first one is the Dynpro processor. All Dynpro programs have flow and processing logic. And it's the Dynpro processor's job to handle the actual flow logic itself. It responds to the user's interaction and controls the further flow of the program. Depending on the user’s interaction.

What I'm trying to say is it's responsible for dialog control in the screen itself, but one important thing to remember is it cannot perform calculations. It's just there to manage the flow logic of a program.

The next important element is the ABAP processor. And this is responsible for the actual processing logic for our programs. It receives screen entries from the Dynpro processor and transmits the screen output to our program. Now it's the ABAP processor that can perform the logical operations and arithmetical calculations in our programs. It can check authorizations and read and write the database over the database interface, and it's the database interface that is our third important element.

Now the database interface is actually a state of ABAP statements that are database independent. Now what I mean by database independent is that we can use a set of ABAP statements that in turn can communicate with any type of database that is being installed, when our system was set up. So whether it's, you know, a Microsoft Sql server or an Oracle database, we can use the same ABAP statements, that is actually called Open SQL to control our database reading and writing over the database interface.