s7.m3.o1 **Overview of Internet technologies and remote use of applications.** V. Casarosa. *CNR-IAT, Via Alfieri 1, 56126 Pisa, Italy.*
Keywords: instrumentation, remote experimental control.

In this lecture we will briefly illustrate the evolution of the networking technology and distributed computing, with the consequent evolution of Internet from an international scientific research network to a worldwide infrastructure to offer and access information and services, and we will overview the technologies underlying the main services available today in Internet.

We will then review some of the main applications available today over the Internet, such as multimedia communications, support of telework and group work, remote access to interactive applications.

s7.m3.o2 **Using the WWW to construct Graphic User Inter-faces to Remote Applications**. A.W. Hewat, *Institut Max von Laue-Paul Langevin, BP 156, 38042 Grenoble Cedex 9, France*.
Keywords: instrumentation, remote experimental control.

It is relatively easy for non-experts to construct simple Graphic User Interfaces (GUI) using the WWW; such interfaces have the advantage that they can be used remotely from any kind of computer, while providing a familiar user-friendly environment. The actual calcul-ations are performed on the more powerful server, where the programmes can be maintained for all, rather than relying on each user to install the latest version.

This is a return to the idea of using terminals on a "central computer", and is a current trend in many areas of computing.

At ILL Grenoble, we use WWW-GUI's to access remote databases, to monitor and even control diffracto-meters, and to execute data analysis programmes on central computers. The most sophisticated of these GUI's use new languages such as Java, but even old Fortran programmes can be re-juvenated on a central server with a WWW interface on the local machine.

The simplest kind of WWW-GUI involves the creation of a "form" using an HTML editor. The user selects from menus on this form, enters parameters etc and submits it to the server, which runs the programme and returns the results to the user. In practice, the server uses a "CGI" script to format the input and output data; this script may be a series of Unix commands, or a complete Perl programme, often calling several other applications to complete the data analysis.

The WWW-GUI form itself can be made more "intelligent" by adding small scripts called "JavaScript" and storing user preferences in local "cookies". For example, Javascript can provide inter-active menus, dimension the elements on the form according to the user's screen size, provide immediate help in the choice of suitable input etc. All of these elements become automatically available to any user connecting to the server.

An additional advantage of this approach is that the GUI sits on top of the original programmes, which need not be changed. Separating the Graphic User Interface from the calculation greatly simplifies programme development, since the calculation can be written in a language such as Fortran or C that can be easily compiled on any machine.

We will demonstrate WWW-GUI's ranging from simple forms for running Lazy PulverIx[1] or drawing VRML crystal structures[2], to the ICSD inorganic data-base server[3], which combines many different applicat-ions. Finally, we will mention a Java interface recently completed by G. Seroul & B. Ouladdiaf for complete diffractometer control.

[1] http://www.ill.fr/dif/lazy.html
[2] http://www.ill.fr/dif/xtal-3d.html
[3] http://barns.ill.fr/dif/icsd/