



Technology for CSP (Consulting Service Providing)

XANDRA® Technology is a software development system designed by the main {GRUPPE} for the production of personalized, individual consulting systems for both intranet and internet.

What is XANDRA® Technology?

XANDRA® Technology is a software development system for the efficient production of high-quality, robust, personalized consulting systems for the internet. XANDRA® Technology provides the complete technical infrastructure required to develop and operate an individual consulting system.

The benefits of using XANDRA® Technology

XANDRA® Technology is an integrated software development system with the following features:

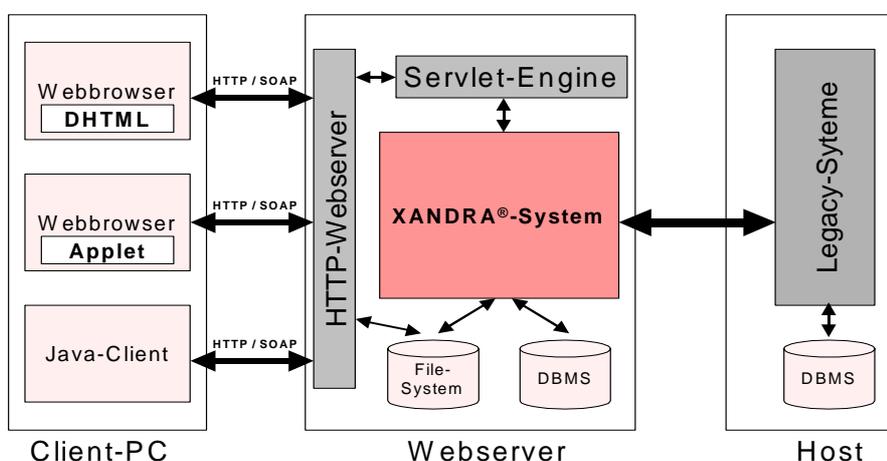
- Innovative technology and the latest implementation techniques
- Good, platform-independent integration into heterogeneous system landscapes
- Flexibility, good scalability and adaptability
- Short development times for appealing, stable interactive consulting systems

Concept and architecture

XANDRA® Technology is based throughout on internet technologies and the benefits they harbour for shared applications on the web. XANDRA® Technology makes a clear distinction between the client and server ends, as well as between presentation and business logic, and data storage. Commands and data are exchanged between the two ends using an internal communication process based on the standard protocol HTTP. Command sequences and data are transmitted in both directions using XML or alternatively SOAP.

At the client end, the display of the user interface and all user interaction take place using a web browser. No other software is required. Browser display is effected by DHTML and Java Script. Complex user interfaces on an intranet can also be displayed using a Java applet or a Java client. The architecture of XANDRA® Technology is so flexible that user interfaces based on DHTML and Java Script as well as applets or Java Clients can all be used at the client end.

At the server end, XANDRA® Technology uses a multilayer architecture with predefined services. Each service has a standardized interface so that any new services can be added as required, while existing ones can be replaced by new implementations.



The above diagram shows the technical infrastructure of XANDRA[®] Technology and an outline of the architecture. The architecture is divided into a number of different layers or tiers. As each tier has precisely defined interfaces and functionalities, individual tiers can be replaced by new implementations without affecting the architecture as a whole.

Client-side technology

Introduction

As XANDRA[®] Technology is consistently based on internet technology, applying it enables you to use all the technologies customarily employed on the web. When developing the client end, our aim wasn't to 'replace the wheel'. Instead we always tried to find ways of enhancing common standard methods.

The results are a series of *technical highlights* hardly found in this form on the internet, and which make life a whole lot simpler for developers. Rather than recapitulate common procedures in the development of web applications, the aim of this chapter is to describe these invaluable features.

XANDRA[®] Technology technical highlights include:

- ✓ **A completely new type of communication technology between client and server**
- ✓ **A totally new type of session management**
- ✓ **Real-time tracking**
- ✓ **New ways of generating HTML user interfaces**
- ✓ **Outstanding user guidance and usability**
- ✓ **Consistent implementation of the thin-client approach**

Lets have a look at these aspects in more detail.

A completely new type of communication technology between client and server

Communication between web clients and the servers has not changed since the early days of the internet:

- Start with an HTML form
- Send it to a script of whatever type (e.g. CGI, ASP, servlets) ...
- ... where the data are evaluated
- The results are sent to the client

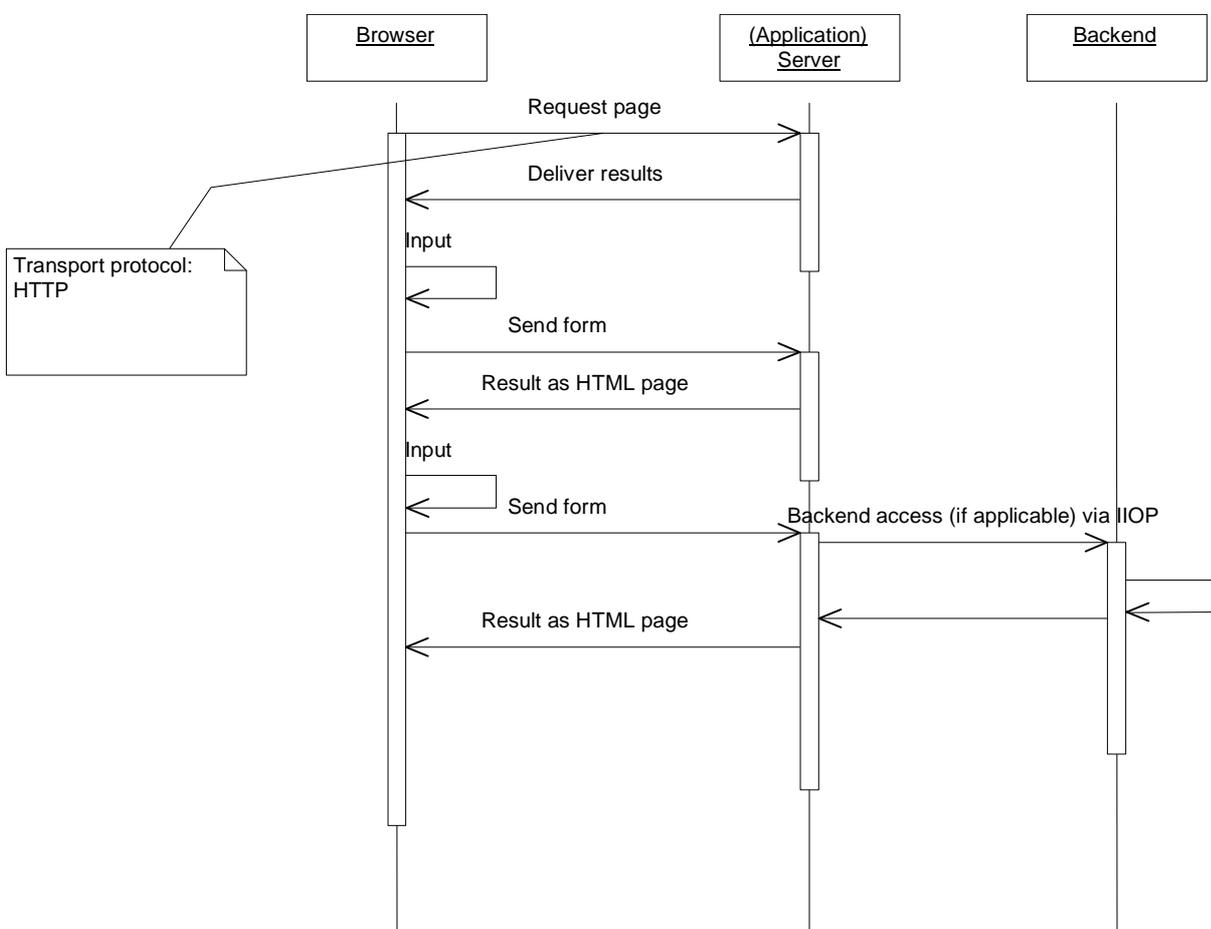
The data on the form are sent to the script as key/value pairs for processing. Drawbacks of complex applications include the following:

- A script or "business object" has to be programmed for each page.
- When backtracking, the 'old' values have to be reconstructed at the server. (For this purpose the data are dynamically entered into the HTML forms, resulting in data being mixed with data presentation.)

- Going to a new page always means contacting the server in order to dynamically build the next page based on the previous data input. This places a considerable strain on the server if it has to deal with several users at the same time.
- If the resulting page closely depends on the data input, the whole process becomes much more complex since each page may result in not just one but several resulting pages.

Even if the server end is skilfully broken down using an MVC (Model View Controller), the result is a confusingly large quantity of small “server objects”. These objects have to be combined within a control instance to form a transaction before sequences can at all be modelled.

The following UML diagram shows the current state of the art.



The technology of the HTML form was simply not designed to relate a complex sequence of interdependent dialogues to each other.

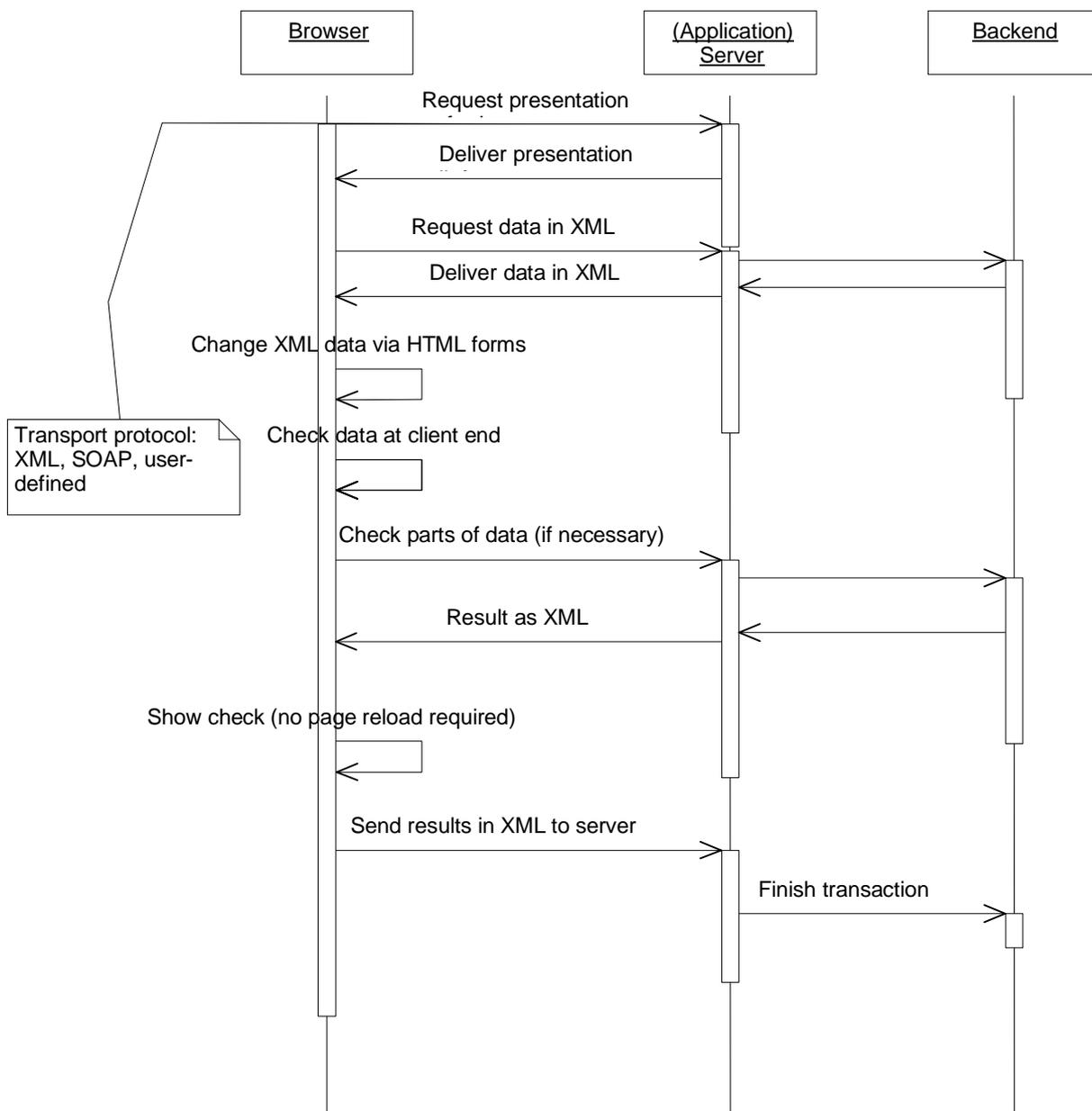
The XANDRA[®] Technology way: using XML as the transport medium

Before going any further, let’s start by emphasizing that XANDRA[®] Technology still allows you to carry on working in the way you’re used to – although you probably won’t want to once you’ve seen how things can be done more efficiently. In XANDRA[®] Technology, communication between the web client and web server takes place as follows:

- You define an XML scheme specifying the data of your dialogue sequence.
- You develop a server-based command which processes all the data of your business process as an XML flow. Alternatively, of course, you can just process parts of the data and embed the command inside a transaction. The important thing is that the server-based commands do *not* refer to your HTML forms. In fact your data do need not be geared to the forms in any way.
- You develop the graphic style of your dialogues for your data using Java Script.
- The data can now be captured in any browser. You don't have to worry about assigning the data to forms, transmitting data or session management at this point. This is all taken care of transparently by the XANDRA[®] runtime system.
- If necessary, *all* the data of your transaction can be sent in XML to the server processing the data.

The elegance of this approach is that *data and presentation are completely separated*, even at the client end. The exchange of data is orientated not towards individual pages but rather *the whole transaction*, minimizing complexity at the server end.

The approach taken by XANDRA[®] Technology is shown in the following diagram:



Note that:

- ✓ XML (SOAP) rather than 'serialized forms' is transmitted between client and server. The conversion of form data into 'business data' is hence eliminated.
- ✓ Results are transmitted to the server 'en bloc' for processing.
- ✓ The number of server hits does not increase with more complex dialogue sequences.
- ✓ Server hits can take place *without the HTML page being reloaded*. This enables for instance complex verification to be carried out within a form, e.g. checking the postcode via a server call.
- ✓ The server can be immediately used as a web service for SOAP-enabled clients.
- ✓ The server can still be addressed via other SOAP architectures and can for example be used as a web service.

Summary: XANDRA® Technology finally means that applications no longer have to be developed like inflated terminal applications.

Totally new type of session management

As HTTP is a stateless protocol, the session in traditional client/server architectures has to be artificially recreated. This can be done in several ways, such as by:

- Cookies
- URL rewriting
- Hidden fields in HTML pages

XANDRA®'s session management is based on cookies or alternatively URL rewriting, but greatly simplifies these methods. Session data are stored at the client end for as long as possible. This:

- ✓ ... considerably decreases the strain on the server
- ✓ ... reduces the complexity of session management at the server end

The saving of session data on the client system takes place completely transparently for the developer, who simply doesn't have to worry about it.

If necessary, the data collected are sent as a SOAP command to the server, which then processes the data. Error messages can be passed back to the client via SOAP without necessitating data reload.

Summary: In terms of session management, XANDRA® Technology behaves like a classical client/server architecture. Session management takes place transparently via the runtime system.

Realtime tracking

XANDRA[®] Technology captures the user's surfing behaviour starting from the *first contact in real time*. The standard version logs the following:

- ✓ Links visited
- ✓ The time spent on each page
- ✓ All data previously entered at this website

Optional parameters which can be logged include:

- ✓ The number of mouse movements per unit of time and page
- ✓ The number of keyboard strokes per unit of time and page

These data can be used to improve your website's acceptance. Realtime tracking is especially efficient if you want to react individually to the user's input. For example, information that the user

- ... is 35 years old
- ... navigates quickly and confidently through your website

... can be used in conjunction with demographic data to conclude with a certain degree of probability, say, the user's average income. XANDRA[®] Technology enables you to offer users in this category specially tailored information or to find out more about their income by asking the right questions *during the first contact*.

If for instance users can be identified by a user ID and password, the data can be stored on the server (without violating data protection laws) and used again immediately the next time they visit your website. Hence XANDRA[®] Technology *learns*, adapting itself more closely to your customers each time they visit your website.

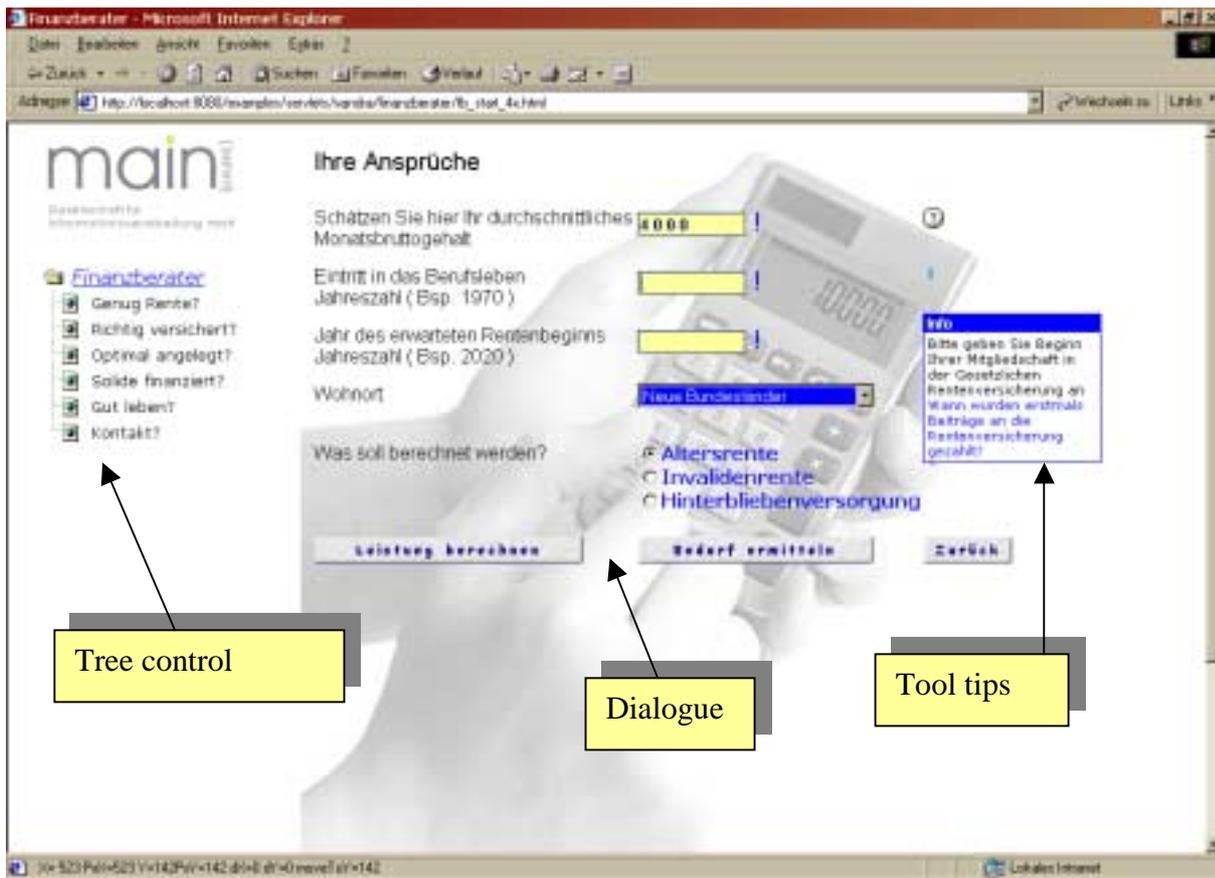
Summary: XANDRA[®] Technology enables you to get to know your customers to a hitherto unknown degree from their very first visit to your website.

New ways of generating HTML user interfaces

XANDRA[®] Technology opens up totally new design possibilities for websites. XANDRA[®] Technology user interfaces are no longer generated in HTML, but instead using a Java Script object model. XANDRA[®] Technology has the largest commercially available library for the production of graphic user interfaces (GUIs). In addition to standard controls, XANDRA[®] Technology gives you:

- A series of new controls enabling completely new forms of user interaction
- Automatic validation of input data fields at the client end
- Automatic production of a "stateful environment" at the client end

The following screenshot contains a few examples of the controls. In practice the objects look like this:



The XANDRA[®] Technology object model is very easy for Java programmers to learn. Two days' training is all you need to start developing your own user interfaces with XANDRA[®] Technology.

The main advantage of this type of GUI development is the simple code modularization. XANDRA[®] Technology enables you to develop reusable components which:

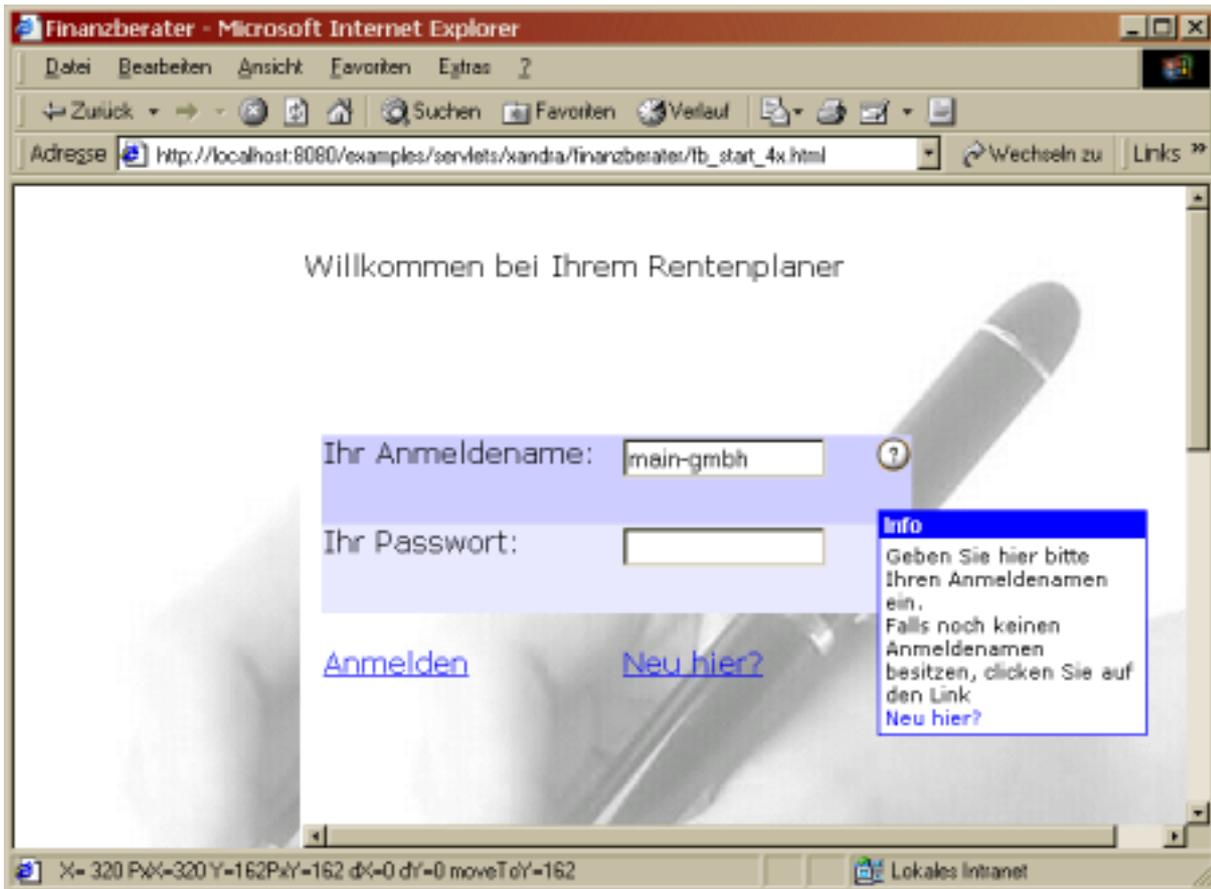
- ✓ ... can be integrated into any XANDRA[®] Technology user interface
- ✓ ... are fully integrated with the backend

XANDRA[®] Technology is available with a host of ready-made components such as a complete login control, which can be integrated into your system with just two lines of code.

For example:

```
g_login_dlg=SECURITY_LOGIN.createLoginDlg("LOGIN",onSendDataDone,onNewUser,
                                     10,10);
g_login_dlg.invalidate();
```

produces the following output:



The login component is immediately connected to the backend system and can be adapted to your own needs.

Summary: *No other commercially available system can provide these features.*

Outstanding user guidance and usability

As you have no doubt already realized, the GUIs which can be produced with XANDRA[®] Technology can be made very user-friendly. Acceptance is achieved through:

- ✓ The consistent use of tool tips
- ✓ Context-sensitive online help
- ✓ Real-time tracking and “learning adaptation to users”. XANDRA[®] Technology enables the whole page to be changed on the fly once a certain user profile has been identified.

Applications of this type are only possible with conventional technology after very elaborate programming.

Summary: *This degree of user-friendliness is rarely encountered on the web.*

Consistent implementation of the thin-client approach

Believe it or not, XANDRA[®] Technology uses neither plug-ins nor Java. XANDRA[®] Technology merely needs a standard browser (Version 4.x or better) with Java Script. Data transfer takes place using the normal HTML standard.

Summary: XANDRA[®] Technology makes for a genuine thin client with 'fat functionality'.

FAQs: XANDRA[®] Technology and client technology

Does Java Script have to be installed on the client to use XANDRA[®] Technology?

Yes, Java Script needs to be activated, otherwise XANDRA[®] Technology cannot work.

Isn't the programming of the user interfaces very elaborate compared to 'regular' HTML?

Don't forget that it's almost impossible to produce XANDRA[®]'s high standard of interfaces using 'regular' HTML. This is because HTML code would have to be provided with plenty of browser-dependent code in order for example to generate tool tips. Therefore we decided to encapsulate the HTML code via an object model and to generate it on the fly at the client end depending on whichever browser is used.

The XANDRA[®] Technology classes were developed with the aim of generating graphically sophisticated interfaces with a simple object model. Developers with a grounding in an object-orientated language like Java can start developing XANDRA[®] Technology user interfaces after just two days of training.

Can XANDRA[®] Technology also work with standard HTML?

Yes, although you won't be able to use all the advantages of XANDRA[®] Technology.

What other benefits does the XANDRA[®] Technology approach have?

- ✓ XANDRA[®] Technology pages are loaded extremely quickly, as only Java Script but no (over-inflated) HTML code needs to be transmitted.
- ✓ XANDRA[®] Technology code can be modularized. The usual approach is to implement recurring graphic details as subprograms. For example, the code for the navigation bar only has to be transmitted once, after which it can be used from the browser's cache.
- ✓ XANDRA[®] Technology comes with a series of *ready-made components* which can be immediately used and connected to the backend system. Hence complete user authorization can be integrated into your page with just two lines of code.

How big is XANDRA[®] Technology?

The basic system consists of about 80 KB of Java Script code. Additional modules can be subsequently loaded via a loader. To minimize the loading time of this 80 KB, the scripts are usually loaded via prefetching in the background when the start page is loaded.

What exactly is "pre-fetching HTML pages"?

'Pre-fetching' refers to loading HTML pages in the browser cache before these pages are actually displayed. Pre-fetching considerably boosts client performance and is supported by XANDRA[®] Technology.

Does XANDRA® Technology need applets?

No, although you can if you like develop your user interface using applets as well.

If I use applets, will the application's structure change?

No, the communication process between client and server is the same in both processes. You simply develop your user interfaces with other means.

Does XANDRA® Technology have problems with firewalls?

No. All protocols are based on HTTP and standard ports.

Is XANDRA® Technology suitable for intranets?

Yes.

Does XANDRA® Technology need plug-ins?

No.

Can I add new GUI elements to XANDRA® Technology?

Yes. The API for producing controls is open and documented.

Does XANDRA® Technology need cookies at the client end?

No, XANDRA® Technology can also work without cookies.

What browsers does XANDRA® Technology support?

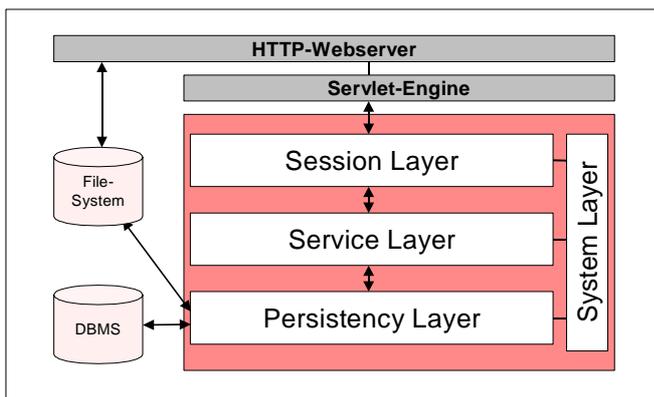
XANDRA® supports Netscape Version 4.x or better, and Internet Explorer Version 4.x or better. Support for Netscape 6.0 is planned. XANDRA® thus covers more than 95% of the browsers used on the internet.

Does XANDRA® Technology support SSL?

SSL support is planned.

XANDRA® Technology and server technology

At the server end, XANDRA® Technology is based on a multilayer architecture. The client sends an HTTP request to the web server. This forwards the query via a servlet to the XANDRA® system, which accepts it and deals with a business process within a transaction. This business process could be a simple database transaction (e.g. "Customer search") or even a complex process such as the preparation of a set of contracts with printing options. The entry point for the XANDRA® core system is always a "worker", which calls up one or more services via a generic command object. This command object describes the business process desired and is responsible for performing the required service.



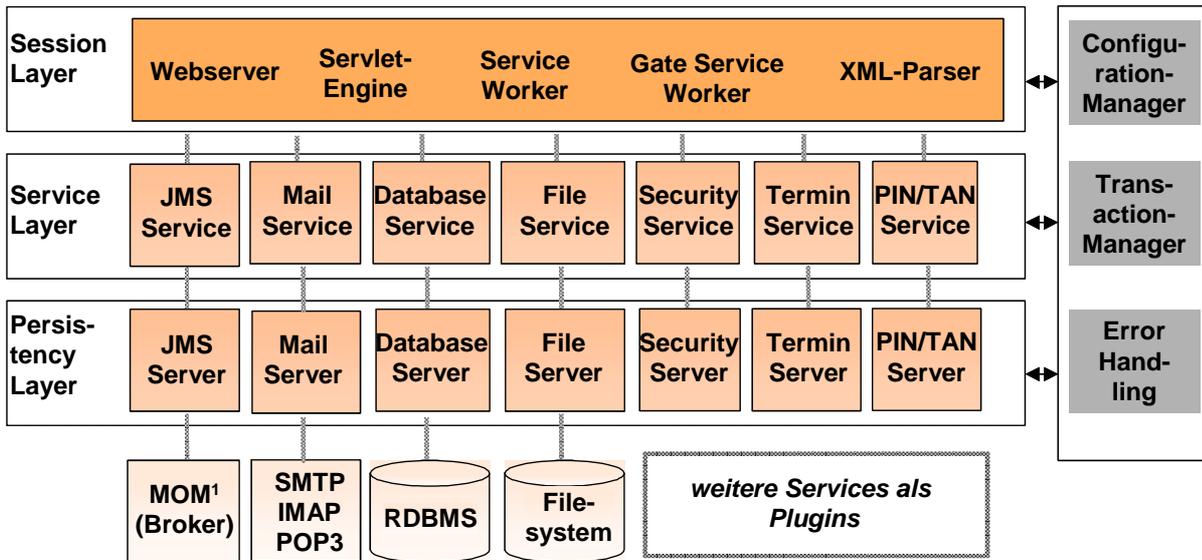
The above diagram shows the basic division of XANDRA[®] Technology into the various levels or layers.

The **SessionLayer** manages and controls the individual sessions together with the application server. All assignments within a session are accepted by a servlet (the central entry point) and delegated to a worker. The worker can call up the various services of the **ServiceLayer** depending on the incoming request and have the assignments carried out. For this purpose the worker generates a command object using the request received and sends it to the relevant service in the **ServiceLayer**. To carry out assignments, the **ServiceLayer** uses the **PersistencyLayer** and the services of the **SystemLayer**. The **PersistencyLayer** contains the access layer for database systems and any other data sources.

In addition to the general system services for the XANDRA[®] system, the SystemLayer also contains the services for logging & tracing via all the other layers. Depending on the settings in the configuration file, the entire output flows are displayed on the monitor or diverted into a file, or even deactivated.

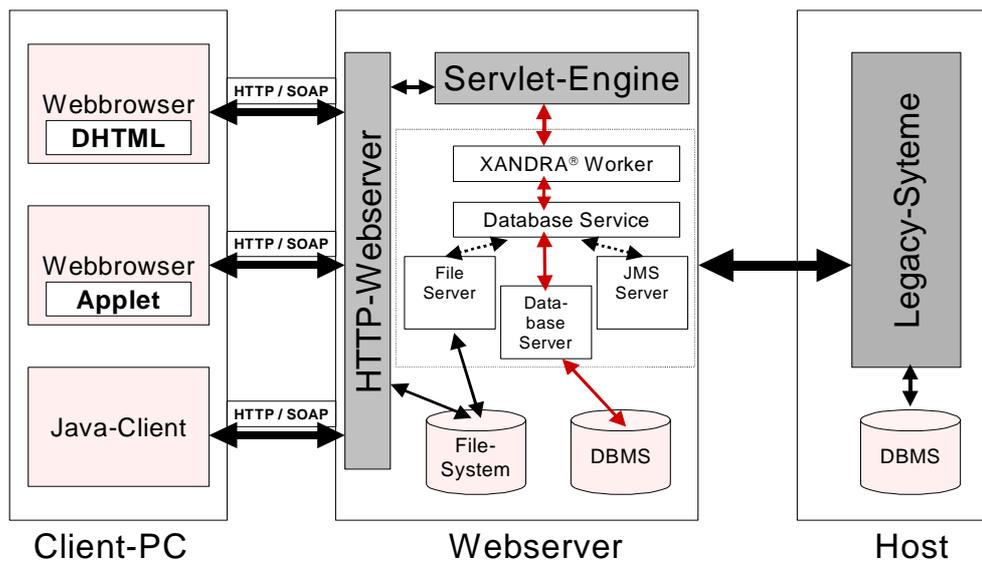
The XANDRA[®] system makes its services available in the form of configurable services. XML-format configuration files are used to control the runtime environment of services. Each service loads the corresponding configuration file during initialization via a configuration manager.

All services (of the ServiceLayer) have a standardized interface, making for a universal, uniform programming model for the usage of the XANDRA[®] system. Each service uses one or more servers for its assignments. For example, the database service uses a database server to access a DBMS. Furthermore, the services can also delegate their assignments to other servers.



As already mentioned above, the strict division between service and servers means that a service can use different forms of server. The database service for example uses in standard cases the database server to perform database operations. The database server geared towards RDBMS can be replaced by a server geared towards an ODBMS. Switching between the two servers as required is possible by means of settings in the configuration file for the database service without having to interfere with the implementation or to recompile the source code. If necessary, queries can also be forwarded by the database service via the JMS server to another system, or dealt with using the file server.

Access to a database system with the XANDRA[®] system then takes place as follows:



The red arrows show the standard route taken by the flow of data via the database service. Alternatively, the system can be configured to divert the data flow via the file server to the file system, or via the JMS server to any other data storage system.

FAQs: XANDRA[®] Technology and server technology

Can I add new services?

The open architecture means that services can be easily implemented and integrated. All that needs to be done is to implement certain interfaces.

Can XANDRA[®] Technology also be integrated into an EJB architecture?

In principle yes. However, the database service may have to be adapted depending on the persistency mechanism (CMP or EMP) and the containers available. Adaptation of the entire architecture to the new EJB specification and the "Connectors" proposed by SUN is planned for mid-2001.

Can I also address the services in one of my own systems?

Yes. The generic interfaces of services mean they can also be directly addressed by other systems.

Requirements for using XANDRA[®] Technology

The low requirements for installing and operating a system based on XANDRA[®] Technology are as follows:

Client end

- A web browser such as Internet Explorer 4.x or better, or Netscape 4.x or better.
- A working internet or intranet connection

Server end

- An installed web server or an application server for Java environments
- Java runtime environmental with a JDK 1.2 or higher servlet engine
- An installed and configured XANDRA[®] system

The advantages of using XANDRA[®] Technology

- Efficient, reliable development of high-quality individual consulting systems
- Flexible interfaces with your existing databases and legacy systems
- Good expansion and adaptation possibilities thanks to multilayer architecture
- Low procurement and migration costs due to platform independence
- Your existing technical infrastructure can still be used – or a new one can be set up in line with your requirements (e.g. web server or application server)