
ELEMENT4SOLUTION



Java Development Framework for Applications

The aim of E4S is to develop:

- Web enabled
- Database driven
- "Rich Client" interfaces



The team of E4S was engaged in several software projects since Java 1.0, using different techniques.

This experience resulted in the fact to establish a development tool to speed up the whole development process.

E4S targets the following topics:

- Application framework, including common used functionality
- Database to Code integration (named fields, database meta schema and generated classes)
- HTML rendering
- Ajax
- Localization
- CSS Integration and Editing

Benefits in the use of E4S:

- Reducing development time & cost
- Build up reliable applications
- Build up state of the art applications

E4S is designed to be used for Java developers building up rich client web-applications.

There is no special or deeper knowledge required, our frameworks will guide you through the development process.

Client's Requirements:

- Web Browser
- Internet Connectivity

Developer's Requirements:

- Fundamentals of Java programming
- HTML, CSS Understanding
- FORMS/CGI Understanding

Server's Environment:

- Apache 2
- Tomcat 5.x
- JDBC compliant database (e.g. MS-SQL Server)

Application Examples 1/2

Includes all HTML Standard tags, but furthermore complete elements like Trees, List and complete database access routines for retrieving and editing.

The screenshot shows a web browser window titled 'door2rental 1.02a - Microsoft Internet Explorer'. The address bar shows 'http://localhost/d2r/d2r?ALIAS=403&TIM=21fc'. The application interface includes a navigation menu on the left, a central table of projects, and a form on the right for editing a project.

Callouts point to the following features:

- Menus**: Points to the top navigation bar with tabs like 'Control-Center', 'Detail-Menu', and 'Vorschau'.
- Database Retrieval**: Points to the table of projects.
- Forms and Validation**: Points to the form fields on the right, such as 'ID', 'Name', and 'Anmerkung'.
- Tree Navigation**: Points to the left-hand navigation tree.
- Sorted Lists**: Points to the table of projects.
- Image Selectors**: Points to the 'Logo' field in the form, which shows a selected image 'images/messen/'.

ID	NAME		
GAST	Alles für den Gast		
CASA	Ambiente & Möbel		
BAUEN	Bauen & Energie Messe		
BW	Bauen & Wohnen		
JAGD	Die Hohe Jagt & Fischerei		
FERIEN	Ferien Messe		
ITNT			
JASPOWA	Ja...a & Fischerei & Offroad		
WOHNEN	W... Interieur		

Logo: BAUEN & ENERGIE MESSE

images/messen/ [X]

Alle Produkte inkludieren?

angelegt: []

modifiziert: []

Application Examples 2/2

Build your own reusable elements.

The screenshot shows a web browser window displaying a web application. The browser title is "door2rental 1.02a - Microsoft Internet Explorer". The address bar shows "http://localhost/d2r/d2r?ALIAS=403&TIM=44b2". The application interface includes a navigation menu on the left with categories like "Funktionen", "Projektdateien", "Hotels", "Artikeldaten", "Benutzer", "Reservierungen", "Auswertungen", "Stammdaten", "Buchungsdaten", and "Datenprüfung". The main content area is divided into sections for "Artikel" and "Projekte". The "Projekte" section contains a table with columns for "BAUEN", "BW", and "CASA". A callout box labeled "Framesets" points to the top navigation bar. A callout box labeled "Grid Editing Elements" points to the table in the "Projekte" section. A callout box labeled "Navigation" points to the left sidebar menu. A callout box labeled "Popups" points to a cell in the table containing the text "MT_01_FOYER/BAUEN".

Artikel		Projekte		
		BAUEN	BW	CASA
		Bauen & Energie Messe	Bauen & Wohnen	Ambiente & Möbel
1	2	3		
1	2	3		
LT_01_10	Logo-Teppich 10 Stk	11	<input type="checkbox"/>	<input type="checkbox"/>
LT_01_5	Logo-Teppich 5 Stk	12	<input type="checkbox"/>	<input type="checkbox"/>
MT_01_FOYER	Mega-Transparent - zentral im Foyer A	13	<input checked="" type="checkbox"/>	<input type="checkbox"/>
MT_02_MALL	Mega-Transparent - Beginn der Mall	14	<input type="checkbox"/>	<input type="checkbox"/>
MT_03_HALLENEINGANG	Werbetransparent über Halleneingang	15	<input type="checkbox"/>	<input type="checkbox"/>
OF_01	Logo im Orientierungsfolder	16	<input type="checkbox"/>	<input type="checkbox"/>
SP_01	Spezial-Werbeträger - bis 3m²	17	<input type="checkbox"/>	<input type="checkbox"/>
SP_02	Spezial-Werbeträger - ab 3m²	18	<input type="checkbox"/>	<input type="checkbox"/>
ST_01	Werbetransparent in Halle A - 2 Seiten	19	<input type="checkbox"/>	<input type="checkbox"/>
ST_01_4	Werbetransparent in Halle A - 4 Seiten	20	<input type="checkbox"/>	<input type="checkbox"/>

Foundation Elements

Basic elements to generate HTML output in a very simple way.

All elementary HTML tags can be used to build own output or new elements. Beside, there are elements with a higher level of complexity to be integrated (e.g. a table that supports interactive sorting).

```
public void HelloWorld( HTML html )
{
    BODY body = html.BODY();
    TABLE table = body.TABLE();
    table.setBorder();
    TR tr = table.TR();
    tr.TD().print("Hello");
    tr.TD().print("World");
}
```

Results in:

```
<HTML>
<BODY>
<TABLE BORDER="1">
<TR>
<TD>Hello</TD>
<TD>World</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Extended Elements

Extending elements, based on HTML foundation elements, make your live easier.

We include some common required functionality in E4S to build up more applications with less effort.

The behaviour of those elements can be changed using Java code using AEP¹⁾ and the look can be changed using **CSS Definitions**.

This includes:

- Tree Navigations
- Menus
- Sortable Tables
- Multi-Value fields, Multi-Click Buttons
- Pickup Value Chooser (e.g. Calendar)
- Input Validating
- Complete **Database Input Procedures**

AEP¹⁾ = Application Exit Points

Application Framework

*Common
used
modules are
included.*

We expect, that every WEB application might need similar functionality, so we have included a set of modules and database tables to cover those functionalities:

- Client Maintenance
- User Maintenance (Group, Company, User) and Login (incl. Password forgotten function)
- Permission Maintenance (Group, Company, User level)
- Online editing of CSS
- Online editing of Translations
- Messages
- Standard Graphics (Icons)
- Performance Statistics
- File selection (similar to Windows™ Explorer)
- Definition of system properties
- System-, Client-, User-, Session- Values & Properties
- Database Metaschema
- SQL Report Generator
- eMail Functionality

Method Reflection (~ Injection)

Ensure integrity between rendered HTML and your Java code.

„Method Reflections“ are the fundamentals of the E4S framework.

A variable (which will be assigned automatically) stands as placeholder for a function in your Java application. This variable is associated with a Java function, wrong references are detected immediately during compile time. The variable can be used as reference on anchor tags (<A>) or form elements <FORM> as well as many other usage within the generated HTML code. By this mechanism, the HTML code becomes secure and only the compiler is the instance to check integrity.

```
public static MethodReflection MyFunct = null;

public void MyFunct( HTML html )
{
    html.println("you clicked");
}

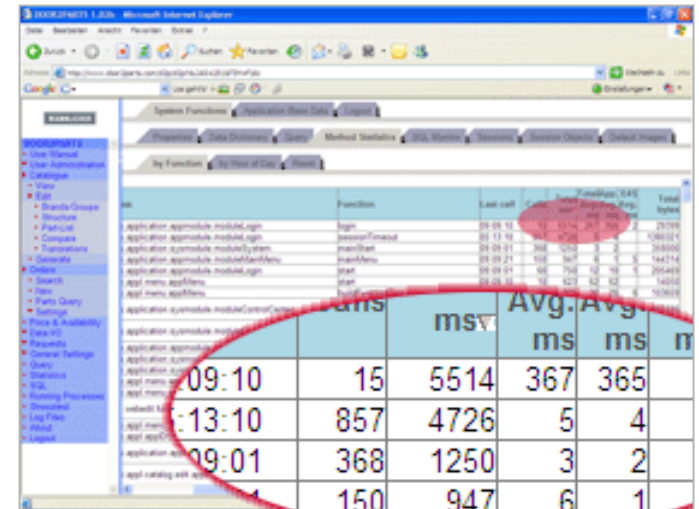
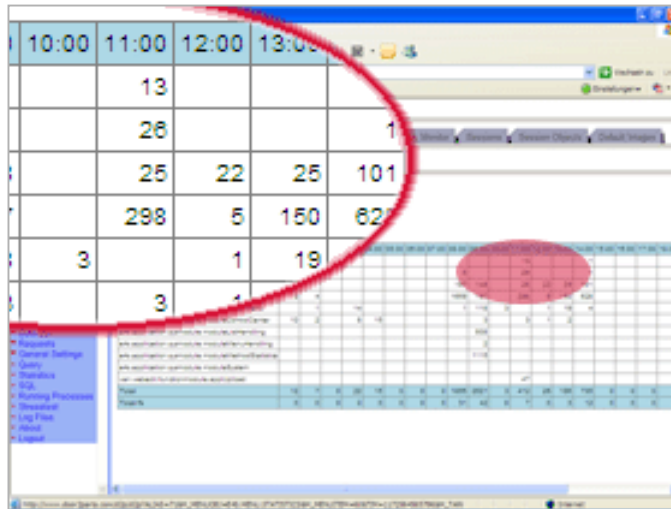
...
// this is rendered in HTML but when the user clicks on
// that anchor-tag, our function MyFunc will be executed.
html.A(MyFunct).print("Click Here");
```

Performance Statistics

Can you say today, which time is consumed in which function?
How often is that function called?
When was this function called?

Each **Method Reflection** (which stands for a user function, grouped in application classes/modules) is being tracked:

- Number of calls (since server restarting, per hour)
- Average Time
- Number of Calls
- Bytes transferred to the client (maximum, average)



Database Metaschema

Database metaschema generates Java classes and ensures database and code integrity.

E4S uses JDBC for database connectivity, a database like mySQL, DB/2, MS-SQL Server or Oracle is additionally required.

Define your data types and tables in the E4S metaschema which is translated into SQL commands to be stored transparency in the database.

Defined data types (e.g. ArticleNumber) stand for entities which are now transparent to the database and to the source code based on an automatically generated Java class. Each time the application uses a defined data type, restrictions like length can immediately be checked to avoid later problems on SQL statements.

Table selection and access classes are also generated out of the meta database, also export/import functionality for data. The E4S datadictionary also contains functionality for direct SQL access and schema comparison. This enables you to run distributed systems on different, slightly incompatible, database products.

The metaschema also includes **Client Distinction** and **Freefield Definitions**.

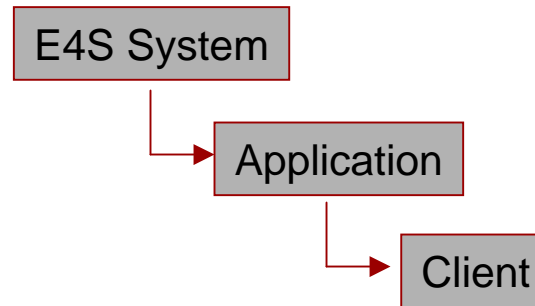
Client Distinction

Designed for Application Service Providing (ASP) business models, we expect that you want to develop one application and share it between different users from different companies, each having their own distinct data.

The datadictionary can generate a column into each table which serves as distinction of data between different clients. On read and write access to that table, the right client is automatically retrieved out of the user's login.



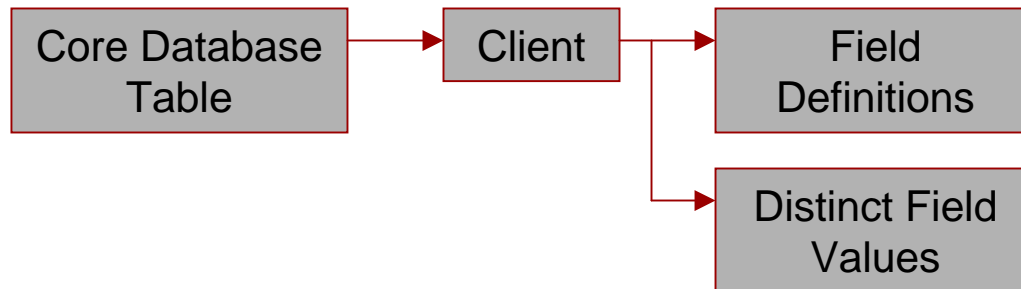
CSS style sheets and file-uploads are also bound on several levels:



E4S includes some system wide **CSS Definitions**, those can be "overwritten" on application level, but furthermore they can be "overwritten" for each client to bring in the CI/CD of your clients.

Freefield Definitions

The E4S metaschema enables you to define fields outside the core datadictionary for one client, so each client can have it's own database fields. Then you can make individual modules for particular clients, targeting those fields. Your main database schema keeps unchanged, compatibility is ensured. The lack of this mechanism is a little bit more requirement in resources. Freefield Definition requires a unique index in the core table.



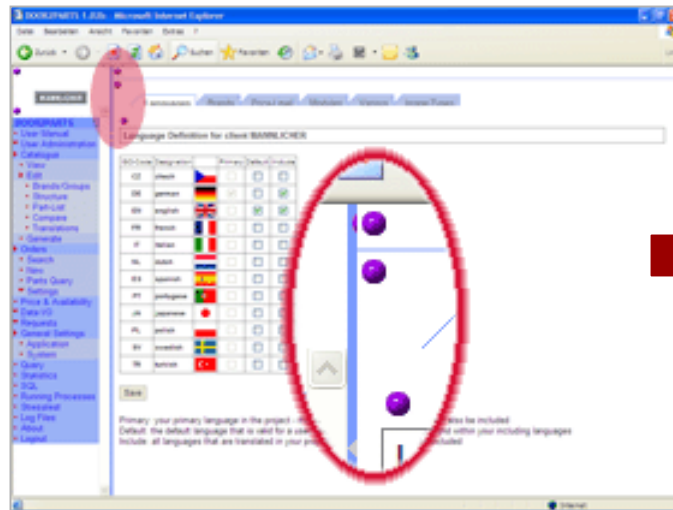
(This mechanism is additional to the core client distinction each table might have).

Remember the old days: You build an application, designed well, but then for one client you need a field in one of the database tables. You won't change the dictionary so you choose a "unused" field. Ugly code because naming gets a different meaning.

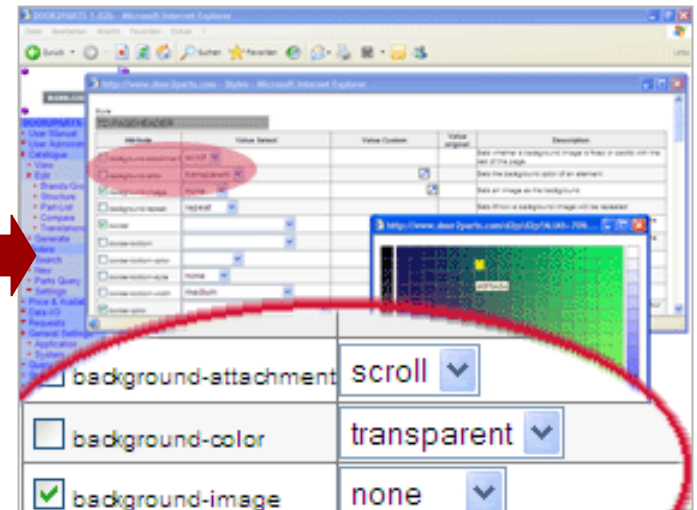
CSS Definitions

Software developers shall develop Java Code, and web designers shall make it colourful.

The Java code covers the applications functionality but also defines names for CSS elements. Later in the software development cycle, those CSS elements can be edited by the web designer (file.css) or simply by a user, even he or she is not common in using CSS attributes.



When CSS editing is turned on, a bubble in each frame provides to edit only the attributes used within this particular frame / page.

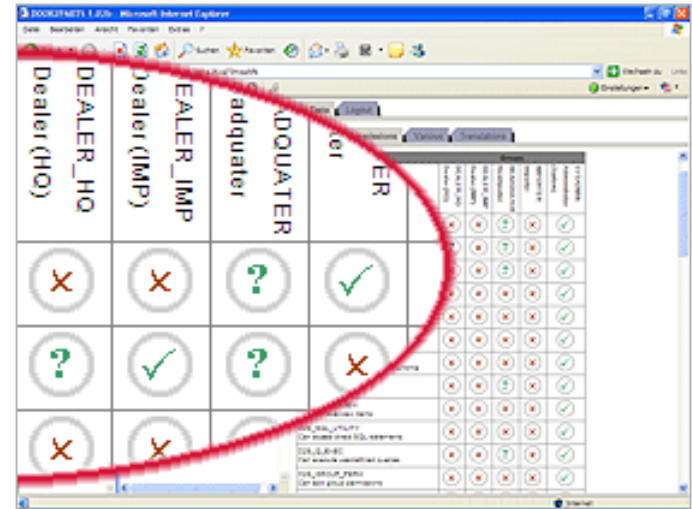
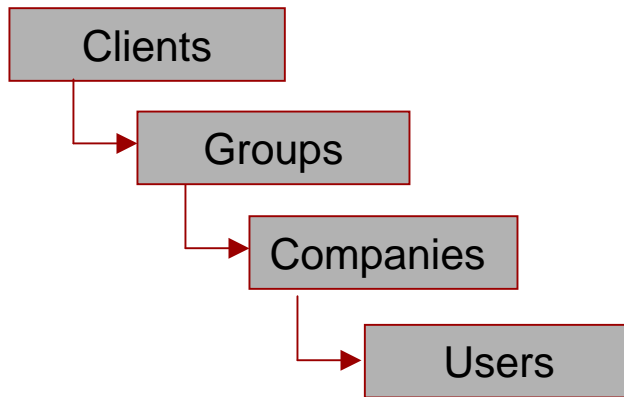


Attributes can be edited by the built in editor to enable less experienced users the modification the layout.

Permissions & Login

Permissions are required in many applications to distinct functionality for different groups of users, e.g. the administrator

E4S includes built in functionality for user management, including permissions. Each permission is given a name by the application developer and can be assigned to a group, a company or a user. It can be queried very simple in your program code.



```
...  
Permission PDEL = new Permission("DELETE.ALL");  
...  
if (! canDo(PDEL))  
{  
    html.Message(Message.ERROR, "No Permission, sorry");  
    return;  
}  
...
```

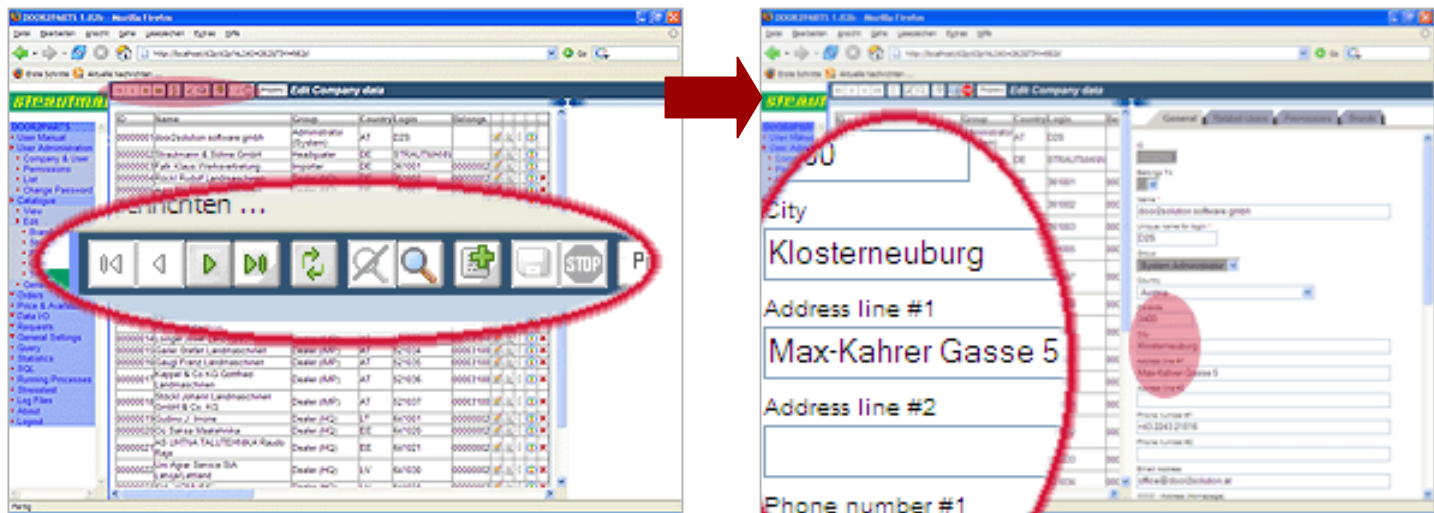
Data Retrieval & Input Functionality

20% to 50% of the application development is required for providing editing functionality for database tables.

This is one of the most powerful tools in E4S.

Based on the database metaschema, E4S can automatically provide functions for data retrieval (list, sort, search) and modifications (insert, update, delete). Data distinction on client level is ensured.

Using AEP¹⁾ technique the application developer can modify the behaviour of these functionality, e.g. selection boxes, validations or hooked in functions.



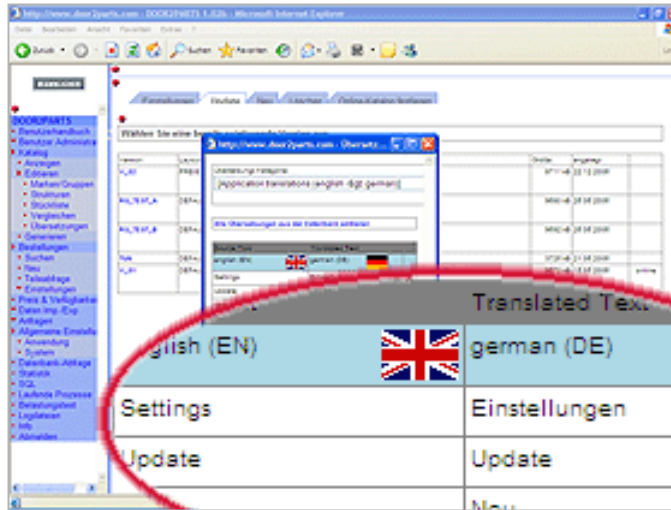
AEP¹⁾ = Application Exit Points

Localization Support



E4S is an European development - maybe this is why a working translation mechanism became very important to us.

An E4S application is developed in one core language (e.g. English). After compilation, the application runs in this language but can be translated, no compilation is required afterwards.



The online translations editor enables users to add or change translations at runtime of your application. Translations can also be synchronized between different E4S applications.

The system language of E4S is English, most elements are translated to German.

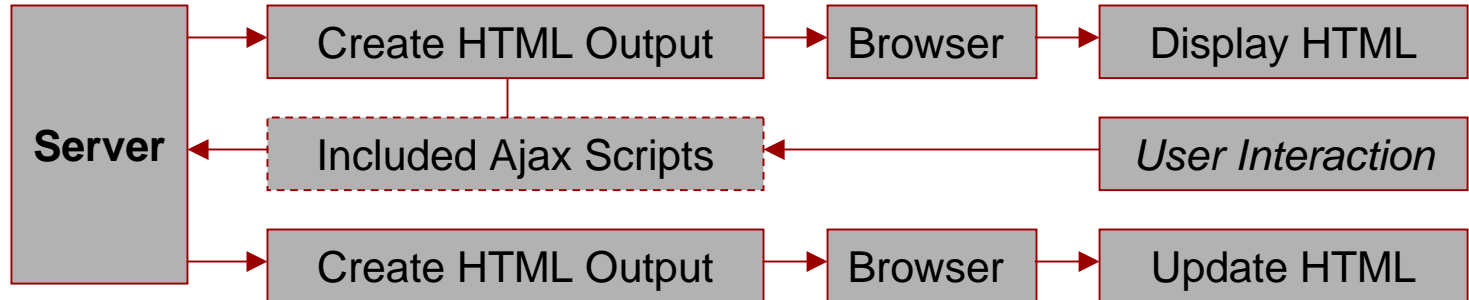
```
...  
html.print("Click here to continue");  
html.translate("Click here to continue");  
// or  
html.print(new LabelApp("You have # new mails",getNumMails()));  
...
```

Unicode support is available, but still during Beta test now.

AJAX

Build dynamically sites using Ajax without the requirement to be an Ajax expert yourself.

Ajax (Asynchronous JavaScript and XML) is a technique to change portions of a web site only. It is covered in E4S as foundation elements, based on simple tags (e.g. <TABLE>, <TR>, <TD>, ..) which become rendered during execution in the client browser's DOM (document object model).



For an example, refer to the E4S tutorial

Error Handling & Debugging

The topic that nobody wants to talk about...

Throwables:

If it happens, that your application throws an exception, this will be displayed on the user's screen but also an email will be sent to an administrator one time a day the same error occurs.

Debugging:

We do not provide debugging, as our applications are servlets and debugging becomes very tricky and difficult. But what we can support you are many information about user processes online, such as transparent session objects.

Tracing:

Simple, but efficient, E4S has included a tracing mechanism that let's you retrieve information at runtime about variables or objects in the server's log files together with the line number of your application.

SQL:

You can have SQL access to your databases via the HTML environment. This becomes important when you deal with several distributed applications, cross platform and cross databases (e.g. when running on an AS/400 using DB2).

Contact Information

*Please
contact us if
you have
any
questions or
you want a
personal
online demo
of E4S.*

www.element4solution.com



door2solution software GmbH
Max-Kahrer Gasse 5
3400 Klosterneuburg
AUSTRIA

office@door2solution.at
www.door2solution.com
+43 2243 21816

Windows, MS-SQL Server are registered trademarks of Microsoft
Java is a registered trademark of Sun Microsystems
element4solution is a registered trademark of door2solution