

Welcome to PHOENIX CONTACT

Web-Technologie in der Automatisierung

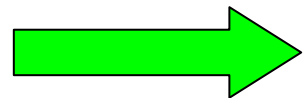
Systemvorteil zum Nutzen
des Anwenders



Web-Technologie in der Automatisierung

Systemvorteil zum Nutzen des Anwenders

- Diagnose von Geräten
- Parametrierung von Geräten und Infrastrukturkomponenten



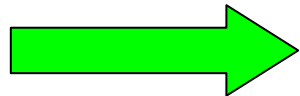
Bedienen und Beobachten mit Webtechnologie!

Bedienen und Beobachten mit Webtechnologie

WARUM?

Stand der Technik

- Bedienen und Beobachten von Anlagen oder Maschinen
 - PC basiert mit SCADA-Systemen (Supervisory Control and Data Acquisition)
 - Bedienpanel verbunden über Feldbus
- Fernzugriffe
 - Modem (Analog/ISDN)



Warum Webtechnologie?

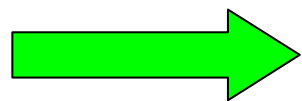
Schwachpunkte der bisherigen Lösungen

- Bedienen und Beobachten von Anlagen oder Maschinen
 - PC basiert mit SCADA-Systemen (Supervisory Control and Data Acquisition)



- Bedienpanel verbunden über Feldbus
→ zu unflexibel

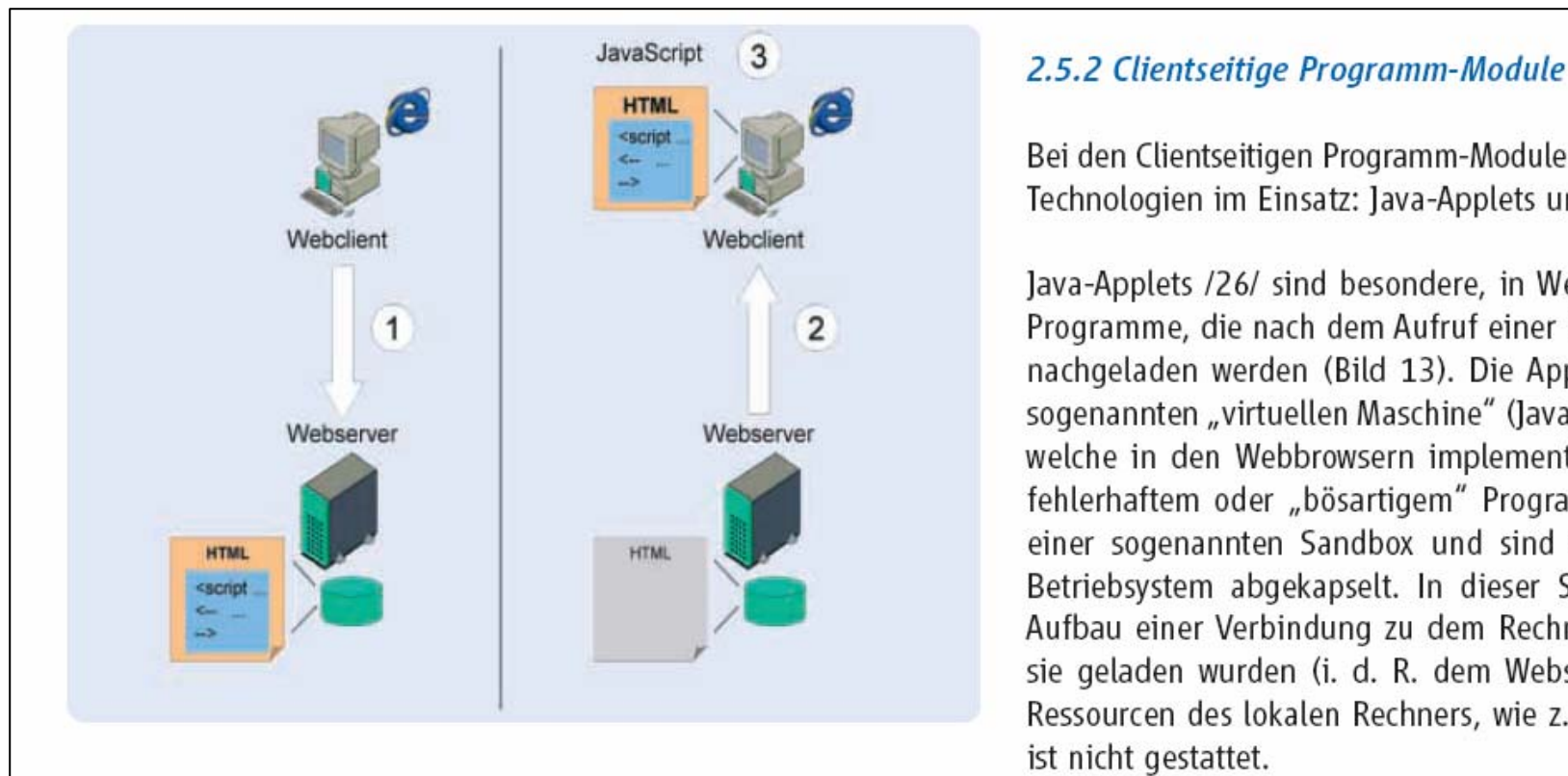
- Fernwartung
 - Modem (Analog/ISDN)



Kostengünstige und flexible Lösungen

Webbezogene Architekturen

Visualisierung von Prozessgrößen



2.5.2 Clientseitige Programm-Module

Bei den Clientseitigen Programm-Modulen sind verschiedene Technologien im Einsatz: Java-Applets und

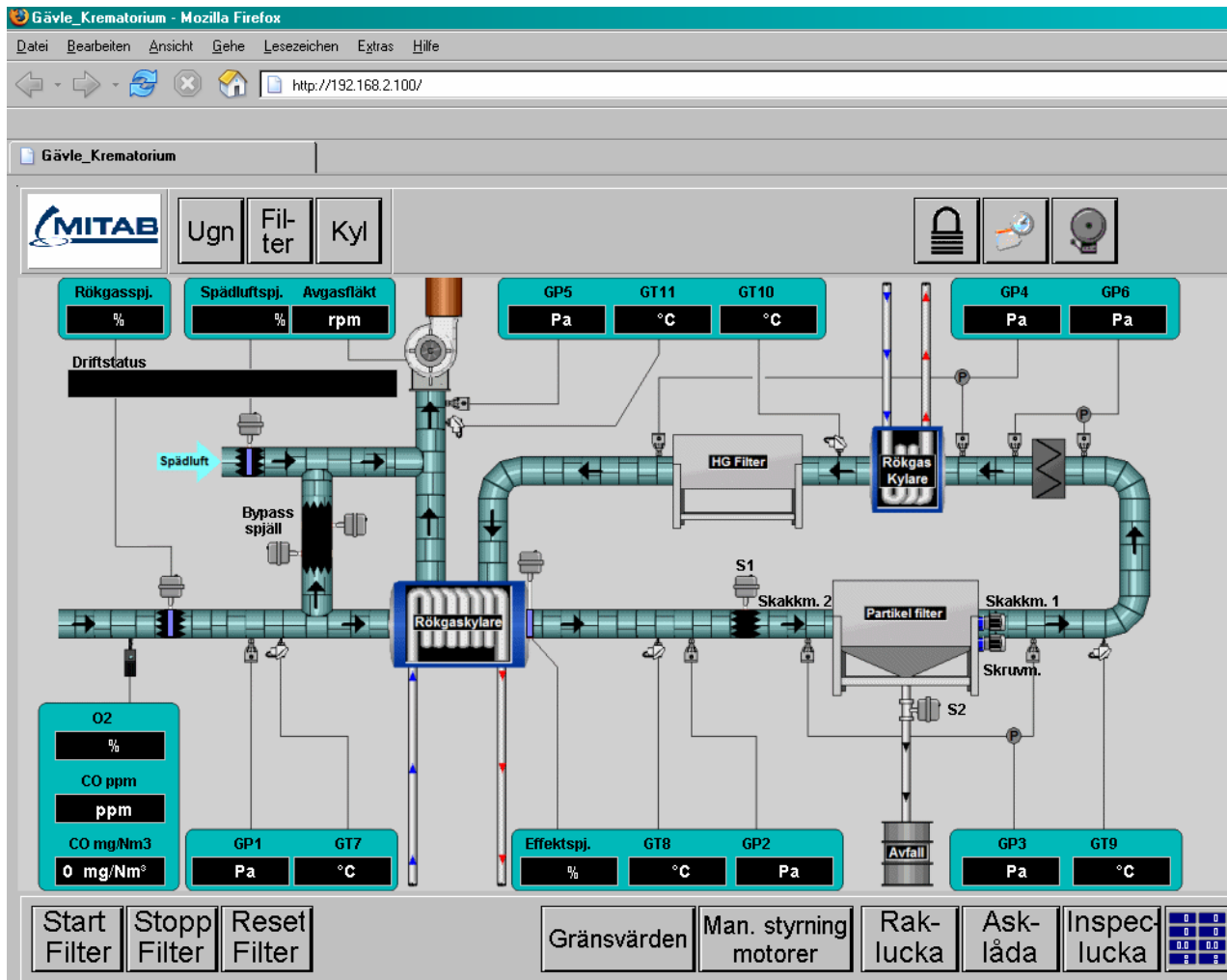
Java-Applets [26] sind besondere, in Webbrowsern ausführbare Programme, die nach dem Aufruf einer HTML-Seite nachgeladen werden (Bild 13). Die Applets werden in einer sogenannten „virtuellen Maschine“ (Java Virtual Machine) ausgeführt, welche in den Webbrowsern implementiert ist. Applets sind von einem fehlerhaftem oder „böartigem“ Programm getrennt und sind in einer sogenannten Sandbox und sind vom Betriebssystem abgekapselt. In dieser Sandbox wird der Aufbau einer Verbindung zu dem Rechner des Benutzers geladen wurden (i. d. R. dem Webserver). Die Ausführung von Ressourcen des lokalen Rechners, wie z. B. die Ausführung von Applets ist nicht gestattet.

Webbezogene Architekturen

Visualisierung von Prozessgrößen

	<i>serverseitige Script-Engine JavaScript im Webbrowser HTTP</i>	<i>serverseitige Script-Engine JavaApplet im Webbrowser / Flash-Video im Webbrowser HTTP</i>
<i>Merkmal</i>	<i>Bewertung / Anwendungsaspekte</i>	<i>Bewertung / Anwendungsaspekte</i>
<i>Erstellungs- und Nutzungsaspekte</i>	<i>Erstellungs- und Nutzungsaspekte</i>	<i>Erstellungs- und Nutzungsaspekte</i>
Entwicklung	Kenntnisse in HTML, JavaScript und der serverseitigen Scriptsprache	Kenntnisse in HTML, Java / Flash und der serverseitigen Scriptsprache
Engineering (Systemintegration, konfiguration und pflege)	Kenntnisse in HTML, JavaScript und der serverseitigen Scriptsprache. Netzwerkkennnisse	Kenntnisse in HTML, Java / Flash und der serverseitigen Scriptsprache. Netzwerkkennnisse

Kundenapplikationen



Kundenapplikationen

Gävle_Krematorium - Mozilla Firefox

http://192.168.2.100/

Gävle_Krematorium

MITAB Ugn Filter Kyl

HBK Temp. °C Huvudbrännare % Primärluft % Ugnstryck Pa

O2 % CO PPM ppm CO mg/Nm³ 0 mg/Nm³ Skorstens temp. °C

Ugnsfläkt Öljepump Kompressor

Kremeringsnr. 0 Driftstatus Operatör

Kremering: % 0 Min.

Filterdrift

iltstyrd

EBK Temp. °C Sekundärbrännare % Sekundärluft %

Rökgasspjäll % Avgasfläkt rpm GP1 Pa

Ugn Start Ugn Stopp P1 P2 P3 Krem. Start +5 Min. -5 Min. Kyl ugn Rak-lucka Ask-låda Inspec-lucka

Webbezogene Architekturen

HTML & JAVA

```
index_ger[1] - Editor
Datei Bearbeiten Format Ansicht ?

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3c.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3c.org/1999/xhtml" xml:lang="de" lang="de"><!-- standard_html_header -->
<head>

<link rel="stylesheet" type="text/css" href="http://www.webmation.eu/content/stylesheet_ges.css" />

<title>WEBMATION - WEB in autoMATION</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<link rel="schema.DC" href="http://purl.org/DC/elements/1.1/" title="http://purl.org/DC/elements/1.1/" />
<meta name="DC.Title" content="WEBMATION - WEB in autoMATION" />
<meta name="DC.Title.Alt" content="Home" />
<meta name="DC.Date" content="22.09.2006" />
<meta name="DC.Format" content="text/html" />
<meta name="DC.Identifier" content="content" />
<meta name="DC.Language" content="ger" />

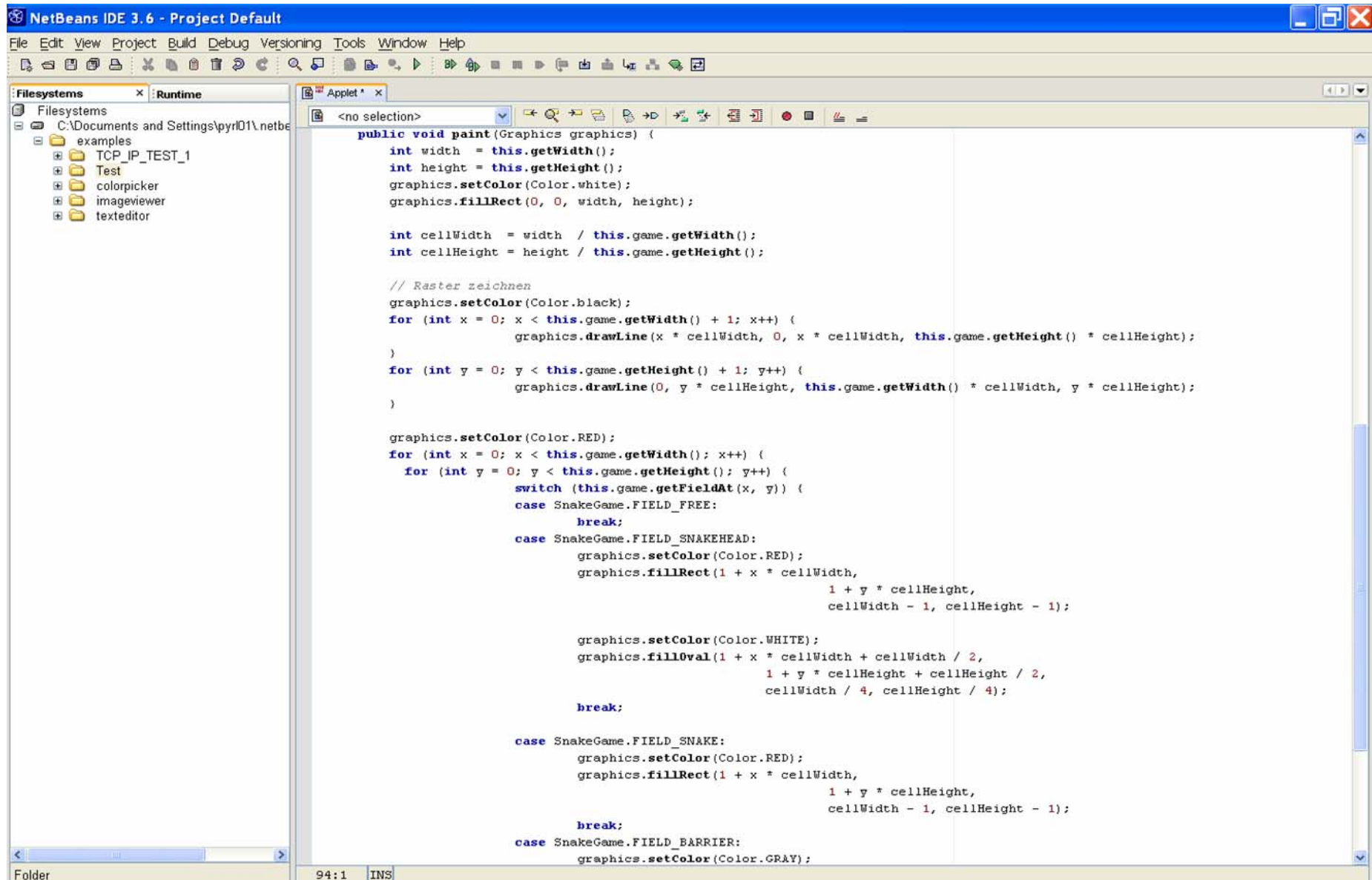
<meta name="DC.Type" content="Home" />
<meta name="navigation" content="2" />

<script type="text/javascript" charset="UTF-8" src="http://www.webmation.eu/content/stylesheet_ges.js"></script>
<script type="text/javascript" charset="UTF-8" src="http://www.webmation.eu/content/comlib_js?lang=ger&manage_lang=ger"></script>
<script type="text/javascript" charset="UTF-8" src="http://www.webmation.eu/content/managerlib_js?lang=ger&manage_lang=ger"></script>
<script type="text/javascript" charset="UTF-8" src="http://www.webmation.eu/content/formlib_js"></script>
<script type="text/javascript" charset="UTF-8" src="http://www.webmation.eu/content/datalib_js"></script>

<!--
<script language="JavaScript" type="text/javascript" src="/lib/js/jquery/jquery.js"></script>
<script language="JavaScript" type="text/javascript" src="/lib/js/jquery/interface/interface.js"></script>
-->
<script language="JavaScript" type="text/javascript" src="/lib/js/aufklappbox.js"></script>
```

Webbezogene Architekturen

HTML & JAVA



The screenshot shows the NetBeans IDE 3.6 interface. The title bar reads "NetBeans IDE 3.6 - Project Default". The menu bar includes "File", "Edit", "View", "Project", "Build", "Debug", "Versioning", "Tools", "Window", and "Help". The toolbar contains various icons for file operations and development. On the left, the "Filesystems" pane shows a tree view of the project structure under "C:\Documents and Settings\pyr01\netbe", including folders like "examples", "TCP_IP_TEST_1", "Test", "colorpicker", "imageviewer", and "texteditor". The main editor window, titled "Applet*", displays the following Java code:

```
public void paint(Graphics graphics) {
    int width = this.getWidth();
    int height = this.getHeight();
    graphics.setColor(Color.white);
    graphics.fillRect(0, 0, width, height);

    int cellWidth = width / this.game.getWidth();
    int cellHeight = height / this.game.getHeight();

    // Raster zeichnen
    graphics.setColor(Color.black);
    for (int x = 0; x < this.game.getWidth() + 1; x++) {
        graphics.drawLine(x * cellWidth, 0, x * cellWidth, this.game.getHeight() * cellHeight);
    }
    for (int y = 0; y < this.game.getHeight() + 1; y++) {
        graphics.drawLine(0, y * cellHeight, this.game.getWidth() * cellWidth, y * cellHeight);
    }

    graphics.setColor(Color.RED);
    for (int x = 0; x < this.game.getWidth(); x++) {
        for (int y = 0; y < this.game.getHeight(); y++) {
            switch (this.game.getFieldAt(x, y)) {
                case SnakeGame.FIELD_FREE:
                    break;
                case SnakeGame.FIELD_SNAKEHEAD:
                    graphics.setColor(Color.RED);
                    graphics.fillRect(1 + x * cellWidth,
                                     1 + y * cellHeight,
                                     cellWidth - 1, cellHeight - 1);

                    graphics.setColor(Color.WHITE);
                    graphics.fillOval(1 + x * cellWidth + cellWidth / 2,
                                     1 + y * cellHeight + cellHeight / 2,
                                     cellWidth / 4, cellHeight / 4);

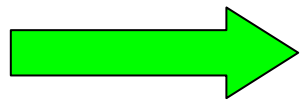
                    break;
                case SnakeGame.FIELD_SNAKE:
                    graphics.setColor(Color.RED);
                    graphics.fillRect(1 + x * cellWidth,
                                     1 + y * cellHeight,
                                     cellWidth - 1, cellHeight - 1);

                    break;
                case SnakeGame.FIELD_BARRIER:
                    graphics.setColor(Color.GRAY);
```

Webbezogene Architekturen

HTML & JAVA

- HTML & JAVA sind für den Anwender für die Erstellung von Visualisierungsseiten zu komplex!
- Schwieriges pflegen der Visualisierungsseiten
- Arbeitsschritte komplett verschieden zu bisherigen Lösungen



Vereinfachung des Engineering durch
geeignetes Tool!

Engineeringtool

Phoenix Contact® WebVisit

File Edit View Project Library Layout Window Help

VC15TM001.teq

ViewCON-6300 @GV.SvsSetA01[1].ShipName DATE: @GV.DMY UTC: @GV.HMS LT: @GV.HMS PANEL:

ALM.No: Man. Supp.: 1 2 3 4 ALM. ACK. HORN RESET

SYS: @GV.SvsS STATION: @GV.SysS WATCH: @GV.SysS DUTY: @GV.SysS BACKUP: @GV.SysS

@GV.WEB@GV.WEB Alarm List[1].Alarm Desci @GV.WEB @GV.WEB Alarm List[@GV.WE @GV.WE @GV.WEB_Alarm_List[1].

ALARM LIST THERMONITOR @GV.Ther @GV.ThermA01[@COFF_PointTM001@

EVENTLOG THERMONITOR LIST

ALARM GROUP

I/O STATUS LIST

TREND LOG

BARGRAPH

GRAPHIC

THERMONITOR

TANK SYSTEM

PMS

SUBSTATIONS

PRINT

BACKUP CALL

MAINTENANCE

Password Mod

userLevel

LOGOUT

HOME

EQUALIZE MAN.

AVERAGE TEMP: @GV.Ther@GV.ThermA01[@COFF_PointTM001@

CYL. ABS. DIV.

1 @GV.T @GV.T
2 @GV.T @GV.T
3 @GV.T @GV.T
4 @GV.T @GV.T
5 @GV.T @GV.T
6 @GV.T @GV.T
7 @GV.T @GV.T
8 @GV.T @GV.T
9 @GV.T @GV.T
10 @GV.T @GV.T
11 @GV.T @GV.T
12 @GV.T @GV.T
13 @GV.T @GV.T
14 @GV.T @GV.T
15 @GV.T @GV.T
16 @GV.T @GV.T
17 @GV.T @GV.T
18 @GV.T @GV.T
19 @GV.T @GV.T
20 @GV.T @GV.T
21 @GV.T @GV.T
22 @GV.T @GV.T
23 @GV.T @GV.T
24 @GV.T @GV.T

TEMP. DEVIATION

ABSOLUTE TEMP.

Engineeringtool

Phoenix Contact® WebVisit

File Edit View Project Library Layout Window Help

VC15H0001.teq

ViewCON-6300 @GV.SvsSetA01[1].ShipName DATE: @GV.DMY UTC: @GV.HMS LT: @GV.HMS PANEL:

ALM.No: Man. Supp.: 1 2 3 4 ALM. ACK. HORN RESET

SYS: @GV.SvsS STATION: @GV.SysS WATCH: @GV.SysS DUTY: @GV.SysS BACKUP: @GV.SysS

@GV.WEB@GV.WEB Alarm List[1].Alarm Desci @GV.WEB @GV.WEB Alarm List[: @GV.WE @GV.WE @GV.WEB_Alarm_List[1].

ALARM LIST HOME

EVENTLOG

ALARM GROUP

I/O STATUS LIST

TREND LOG

BARGRAPH

GRAPHIC

THERMONITOR

TANK SYSTEM

PMS

SUBSTATIONS

PRINT

BACKUP CALL

MAINTENANCE

Password Mod

userLevel

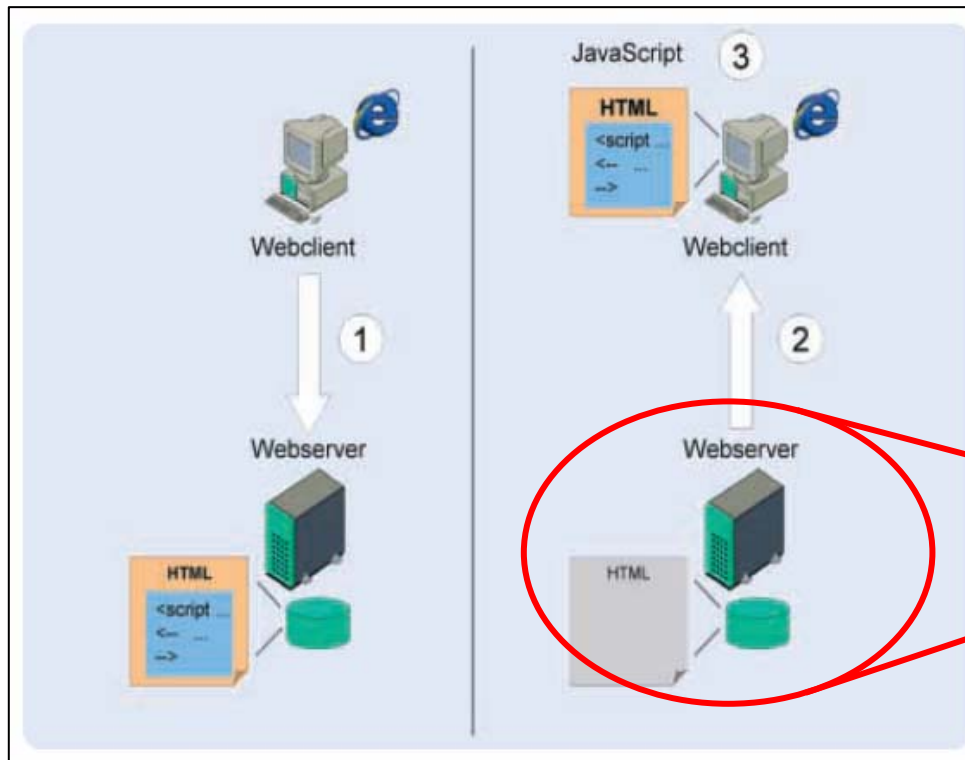
LOGOUT

HOME

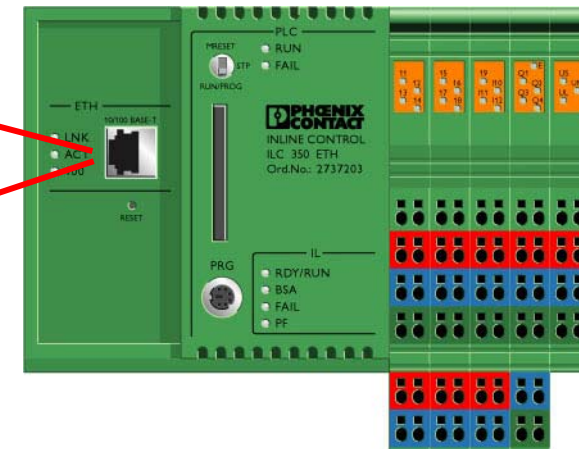
UK

DE

Wo ist der Webserver?



Ein Webserver auf jeder Steuerung!
(Von der Kleinststeuerung bis zur High End SPS)



Zusammenfassung

- Moderne Engineeringtools ermöglichen die Verwendung der Webtechnologie ohne HTML & JAVA Kenntnisse
- In Steuerungen integrierte Webserver kommunizieren direkt mit den Clients
- Visualisierungsclients benötigen lediglich einen Browser und eine JVM (Java Virtual Machine)
- Durch die Verwendung von Webtechnologie lassen sich kostengünstige Bedien- und Beobachtungslösungen realisieren