

## INHALTSVERZEICHNIS

<b>1 EINFÜHRUNG</b>	<b>1</b>
1.1 Komponentenbasierte Softwareentwicklung	1
1.2 Technologie-Trends	2
1.3 Das Ergebnis der technologischen Evolution	3
<b>2 ASPEKTE EINER VERTEILTEN UMGEBUNG</b>	<b>5</b>
2.1 Sichten	5
2.1.1 Das logische Modell	5
2.1.1.1 Geschäftsdomäne	6
2.1.1.2 Geschäftsobjekt	7
2.1.1.2.1 Verhalten	8
2.1.1.2.2 Eigenschaften und Zustände	8
2.1.1.2.3 Beziehungen	8
2.1.1.2.4 Eindeutige Identifizierbarkeit	8
2.1.1.2.5 Persistenz	9
2.1.1.2.6 Ereignisse	9
2.1.1.2.7 Ausnahmebedingungen	9
2.1.1.3 Komponente	9
2.1.1.4 Anwendung	12
2.1.1.5 Die Gesamtschau	12
2.1.2 Das technische Umfeld	13
2.1.2.1 Betriebssystem und Netzwerk-Dienste	13
2.1.2.2 Plattform für verteilte Objekte	13
2.1.2.3 Komponenten-Plattform	14
2.1.2.4 Systemorientierte Frameworks	14
2.1.3 Infrastruktur	15
2.1.3.1 System-Infrastruktur	15
2.1.3.2 Anwendungsorientierte Infrastruktur	16
2.2 Konsequenzen	17
<b>3 ANFORDERUNGEN UND ZIELE</b>	<b>19</b>
3.1 Wirtschaftliche Ziele	19
3.1.1 Steigerung der Entwicklungsproduktivität	19
3.1.2 Integrierbarkeit von Software	19
3.1.3 Verbergen der Heterogenität	20
3.2 Technische Ziele	20
3.2.1 Abstraktion	20
3.2.2 Portabilität und Interoperabilität	20
3.2.3 Transparenz	21
3.2.3.1 Lokationstransparenz	21
3.2.3.2 Persistenztransparenz	21
3.2.3.3 Ausfalltransparenz	21

---

3.2.3.4 Migrationstransparenz	22
3.2.4 Konfigurierbarkeit	22
3.2.5 Ad hoc-Integration	22
<b>4 DAS KONZEPTIONELLE MODELL</b>	<b>23</b>
<b>4.1 Anwendungen und Komponenten</b>	<b>23</b>
4.1.1 Die Steuerung der Anwendungsausführung	26
4.1.2 Die Interaktion mit der Dialog-Oberfläche	27
4.1.3 Die Interaktion mit der Datenhaltung	28
<b>4.2 Konfigurierbarkeit der Komponenten</b>	<b>29</b>
4.2.1 Konfigurationsebenen	30
4.2.1.1 Komponenteninterne Ebene	30
4.2.1.2 Komponentenexterne Ebene	30
4.2.2 Der Konfigurations-Workflow	30
4.2.2.1 Design	31
4.2.2.2 Entwicklung	32
4.2.2.3 Montage und Installation	32
4.2.2.4 Ausführung	32
<b>4.3 Merkmale des konzeptionellen Modells</b>	<b>33</b>
4.3.1 Entkopplung von der Bediener-Oberfläche	33
4.3.2 Entkopplung vom Datenhaltungssystem	33
4.3.3 Flexible Integration von Komponenten	33
<b>4.4 Vorteile des geschäftsobjektzentrierten Ansatzes</b>	<b>34</b>
<b>5 SOFTWARE-ENTWICKLUNG</b>	<b>35</b>
<b>6 PLATTFORMEN FÜR VERTEILTE OBJEKTE</b>	<b>39</b>
<b>6.1 CORBA-Plattform</b>	<b>41</b>
6.1.1 Allgemeine Merkmale	42
6.1.1.1 Herstellerneutralität	42
6.1.1.2 Sprachneutralität	42
6.1.1.3 Portabilität	42
6.1.1.4 Stärken-/Schwächenprofil	42
6.1.2 Besondere Stärken	43
6.1.3 Defizite und Schwächen	43
6.1.3.1 Lokationsinvariante Objekte	43
6.1.3.2 Multiple Schnittstellen	44
6.1.3.3 Pass-by-value-Mechanismus	44
6.1.3.4 Versionenkonzept	44
6.1.3.5 Asynchrone Kommunikation	44
<b>6.2 Java-Plattform</b>	<b>45</b>
6.2.1 Allgemeine Merkmale	46
6.2.1.1 Herstellerneutralität	46
6.2.1.2 Sprachneutralität	46
6.2.1.3 Portabilität	46
6.2.1.4 Stärken-/Schwächenprofil	46
6.2.2 Besondere Stärken	47

6.2.2.1 Plattformunabhängigkeit	47
6.2.2.2 Lokationsvarianz von Objekten	47
6.2.3 Defizite und Schwächen	47
6.2.3.1 Kommunikation zwischen Objekten	48
6.2.3.2 Weitgehende Beschränkung auf Java	48
<b>6.3 COM/DCOM-Plattform</b>	<b>48</b>
6.3.1 Allgemeine Merkmale	49
6.3.1.1 Herstellerneutralität	49
6.3.1.2 Sprachneutralität	49
6.3.1.3 Portabilität	49
6.3.1.4 Stärken-/Schwächenprofil	49
6.3.2 Besondere Stärken	50
6.3.2.1 Homogenität von Spezifikationen und Produkten	50
6.3.2.2 Unterstützung bisheriger Schnittstellen	50
6.3.3 Defizite und Schwächen	50
6.3.3.1 Bindung an Microsoft-Plattformen	50
6.3.3.2 Inhärente Instabilität	50
<b>6.4 Zusammenfassende Betrachtung</b>	<b>51</b>
<b>7 KOMPONENTEN-PLATTFORMEN</b>	<b>53</b>
7.1 CORBABeans	54
7.2 JavaBeans und Enterprise JavaBeans	54
7.3 COM/ActiveX	55
<b>8 SYSTEMORIENTIERTE FRAMEWORKS</b>	<b>57</b>
<b>9 DAS TECHNISCHE UMFELD</b>	<b>59</b>
9.1 Allgemeine Grundsätze	59
9.2 Die Hardware- und Software-Umgebung	59
9.2.1 Anwender-Arbeitsstationen	60
9.2.2 Server-Systeme	61
9.2.3 Kommunikation	61
9.3 Operativer Betrieb	62
9.3.1 Laden von Komponenten	62
9.3.2 Leistungsengpässe und Störungen	63
9.3.2.1 Skalierbarkeit	63
9.3.2.2 Load Balancing	63
9.3.2.3 Verfügbarkeit	64
9.3.3 Technische Wechselwirkungen	65
9.4 Abhängigkeiten	65
9.4.1 Betriebssystem und Netzwerk-Dienste	66
9.4.2 Plattform für verteilte Objekte	66
9.4.3 Komponenten-Plattform	67
9.4.4 Systemorientierte Frameworks	67

<b>10 ARCHITEKTUREN</b>	<b>69</b>
<b>10.1 Anwendungsarchitektur für Komponenten</b>	<b>70</b>
<b>10.2 Die praktische Umsetzung</b>	<b>73</b>
10.2.1 Komponenten der Bediener-Oberfläche	74
10.2.2 Server-Komponenten	75
10.2.3 Kombinationen von Komponentenmodellen	76
<b>10.3 Architektur einer Three-Tier-Anwendung</b>	<b>76</b>
<b>10.4 Wichtige Entscheidungspunkte</b>	<b>79</b>
10.4.1 Isolation von der Bediener-Oberfläche	79
10.4.2 Isolation von den Datenstrukturen	80
10.4.3 Status-Management	81
10.4.3.1 Prozedurale Anwendungssysteme	81
10.4.3.2 Objektorientierte Anwendungssysteme	82
10.4.4 Kommunikation über Ereignisse	84
<b>10.5 Einbettung in die Umgebung</b>	<b>85</b>
10.5.1 Sicherheit	85
10.5.1.1 Schutz der Datenkommunikation	85
10.5.1.2 Schutz der Ressourcen	87
10.5.2 System- und Netzwerk-Management	87
<b>11 TRENDS UND PERSPEKTIVEN</b>	<b>89</b>
<b>11.1 Technologie-Trends</b>	<b>89</b>
11.1.1 Entwicklungs-Werkzeuge	89
11.1.2 Technologien für verteilbare Anwendungen	91
11.1.2.1 CORBA-basierte Technologien	91
11.1.2.1.1 CORBA-Plattform	91
11.1.2.1.2 Komponenten-Technologie	91
11.1.2.1.3 Geschäftsobjekte	92
11.1.2.1.4 Script-Sprache	92
11.1.2.2 Java-Plattform	92
11.1.2.2.1 Java-Plattform	93
11.1.2.2.2 Komponenten-Technologie	93
11.1.2.2.3 Script-Sprache	94
11.1.2.3 COM/DCOM-basierte Technologien	94
11.1.2.3.1 COM/DCOM-Plattform	94
11.1.2.3.2 ActiveX	95
11.1.2.3.3 Script-Sprache	95
<b>11.2 Marktpolitische Trends</b>	<b>95</b>
<b>11.3 Möglichkeiten der Interoperabilität</b>	<b>96</b>
<b>12 RISIKEN UND PROBLEMPUNKTE</b>	<b>97</b>
<b>12.1 Stabilität</b>	<b>97</b>
<b>12.2 Performance</b>	<b>97</b>

<b>12.3 Herstellerabhängigkeiten</b>	<b>98</b>
<b>12.4 Plattformen für verteilte Objekte</b>	<b>99</b>
12.4.1 Java-Plattform	99
12.4.2 CORBA-Plattform	100
12.4.3 COM/DCOM-Plattform	101
<b>13 AKTIONSPLAN FÜR DIE UMSETZUNG</b>	<b>103</b>
<b>13.1 Strategische Entscheidungen treffen</b>	<b>103</b>
<b>13.2 Architekturmodell entwickeln</b>	<b>104</b>
<b>13.3 Werkzeugumgebung bestimmen</b>	<b>104</b>
<b>13.4 Konventionen für Anwendungsdesign und -entwicklung verabschieden</b>	<b>105</b>
<b>14 ZUSAMMENFASSUNG UND AUSBLICK</b>	<b>107</b>
<b>15 ANHANG</b>	<b>111</b>
<b>15.1 Zeitrahmen</b>	<b>111</b>
15.1.1 CORBA-Plattform	111
15.1.1.1 CORBA-ORB	111
15.1.1.2 CORBA Component Model	111
15.1.2 Java-Plattform	112
15.1.2.1 Java Development Kit (JDK)	112
15.1.2.2 HotSpot-Technologie	112
15.1.3 COM/DCOM-Plattform	112
<b>15.2 Ressourcen</b>	<b>113</b>
15.2.1 Spezifikationen	113
15.2.2 Object Request Broker (ORBs)	113
15.2.3 Java-bezogene Ressourcen	114
15.2.4 Komponenten-Anbieter	115