



IBM WebSphere Development Studio Client für iSeries - Produktübersicht

Version 6.0.1 für Windows



IBM WebSphere Development Studio Client für iSeries - Produktübersicht

Version 6.0.1 für Windows

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM WebSphere Development Studio Client for iSeries - Product Overview, Version 6.0.1 for Windows,
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2005
© Copyright IBM Deutschland GmbH 2006

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
SW TSC Germany
Kst. 2877
Januar 2006

Inhaltsverzeichnis

Kapitel 1. PDF-Version 1

Kapitel 2. Übersicht über IBM WebSphere Development Studio Client für iSeries 3

Paketierung für iSeries-WebSphere-Produkte.	6
Zusammenwirken von WebSphere- und Rational- Produkten	6
Paketinhalt	8
iSeries-Entwicklungsroadmap	11
Auswahlmöglichkeiten der Webtechnologie.	13
Lernprogramme und andere Lernmaterialien	15
Neuerungen in diesem Refresh-Pack	17
Übersicht über Tasks	18
Ziel 1: Entwicklung und Verwaltung von Anwen- dungen.	19
Ziel 2: Erstellung von Web-Front-Ends	19
Ziel 3: Erstellung von GUI-Front-Ends	22
Übersicht über Tools	22
Workbench-basierte, integrierte Entwick- lungs- umgebung.	23
iSeries-Entwicklungstools.	23
iSeries-Web Development Tools.	27
iSeries-Java Development Tools.	28

Unterstützung für Struts-Umgebung	29
Entwicklungstools für iSeries-Web-Services	29
Serverentwicklungstools	30
Das IBM WebFacing Tool	30
Integrierter iSeries-Debugger	31
Separat installierbare Funktionen	32
VisualAge RPG	32
CoOperative Development Environment (CODE)	33

Kapitel 3. Vorbedingungen 35

Mit Ihrem HTTP-Server und WebSphere Application Server arbeiten	35
Server-Portnummern	36
Beispielbibliotheken installieren.	38
Beispielbibliothek Ihrer Anfangsbibliotheksliste hinzufügen	39
Eigenes WebSphere Application Server-Exemplar starten	40
Netzlaufwerk dem iSeries-System zuordnen	40

Bemerkungen. 41

COPYRIGHTLIZENZ:	42
Informationen zu Programmierschnittstellen	42
Marken und Servicemarken	43

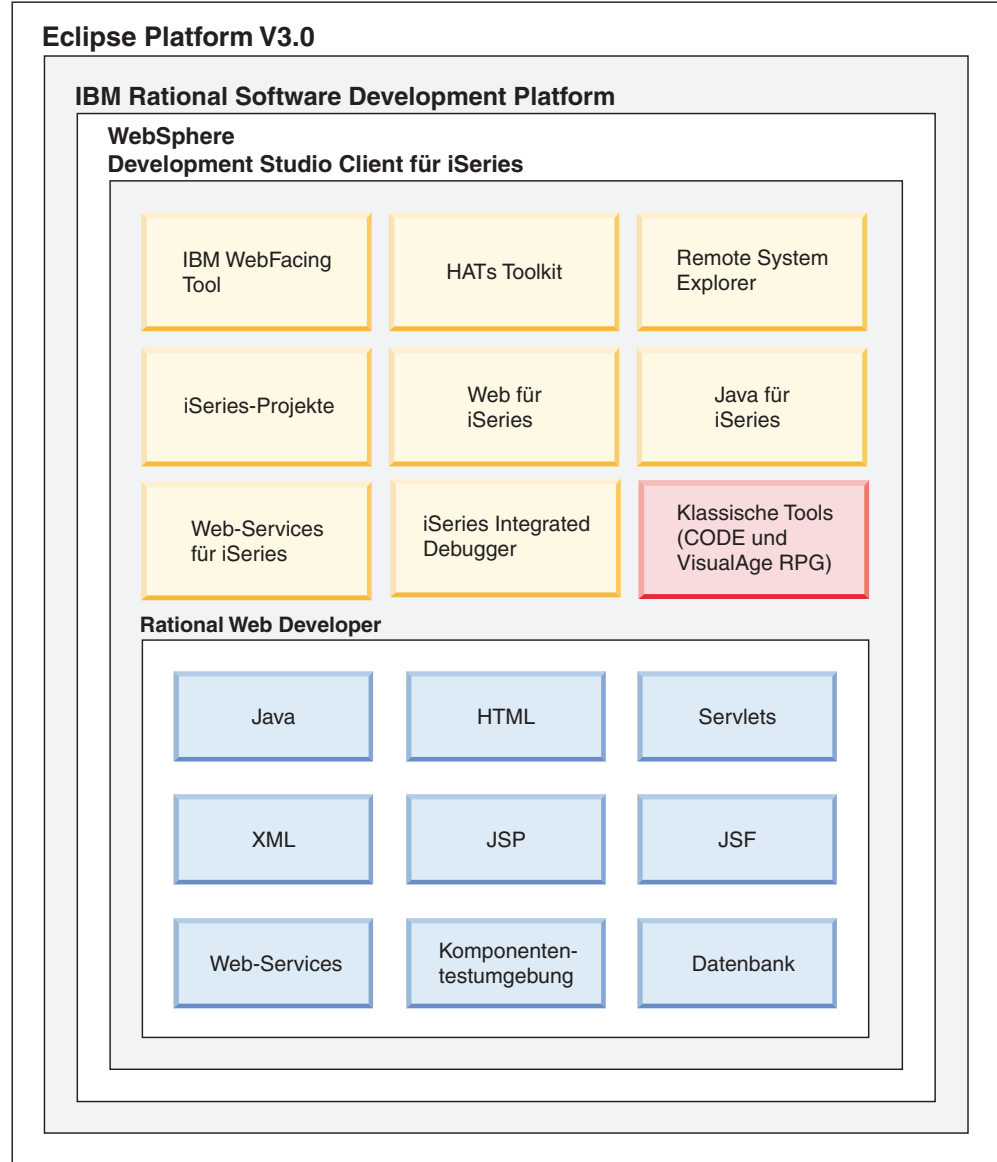
Kapitel 1. PDF-Version

Klicken Sie auf den folgenden Link, um eine PDF-Version der Produktübersicht zu öffnen, die Sie drucken können: [PDF-Version der Produktübersicht](#).

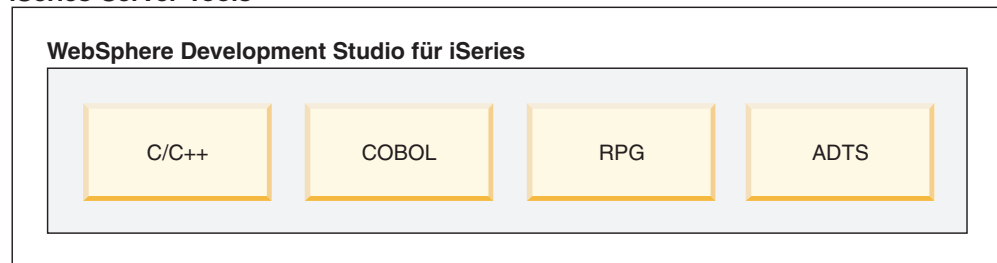
Kapitel 2. Übersicht über IBM WebSphere Development Studio Client für iSeries

IBM WebSphere Development Studio für iSeries stellt ein vollständiges Toolset für den iSeries-Anwendungsentwickler bereit. Das Serverprodukt umfasst die Compiler für RPG, COBOL und C/C++ sowie das Application Development ToolSet (PDM, SEU, SDA und DFU). Die Workstation-Tools, die als WebSphere Development Studio Client bezeichnet werden, stellen Unterstützung für die Erstellung und Verwaltung nativer iSeries-Anwendungen sowie für die Arbeit mit Web-Services und Web- und Java-Anwendungen unter iSeries bereit. Die Workstation-Tools basieren auf dem Programm Rational Web Developer (RWD), das extensive, standardisierte Unterstützung für das Erstellen von Webanwendungen bereitstellt. Eine erweiterte Ausgabe des Produkts schließt alle Funktionen von Rational Application Developer und zusätzlich die erweiterten iSeries-Funktionen, wie beispielsweise Unterstützung für einmalige Anmeldung und EGL, ein. Die folgende Abbildung zeigt die Beziehung zwischen Server- und Client-Tools:

iSeries-Client-Tools



iSeries-Server-Tools



Der Development Studio Client hilft Ihnen dabei, die drei Hauptziele bei der Programmierung von Ihrer Windows(R)-Workstation aus zu erreichen:

Ziel 1: iSeries-Anwendungen entwickeln und verwalten

- Verbindung zu fernen Systemen herstellen und Programmierungstasks ausführen
- Anwendungen (Green-Screen-Anwendungen und/oder webbasierte Anwendungen) editieren, kompilieren, ausführen und eine Fehlerbehebung für diese Anwendungen durchführen
- Mit iSeries-Anwendungen arbeiten, wenn die Verbindung zum iSeries-Server unterbrochen ist
- iSeries-Geschäftslogik in der Programmiersprache Java(TM) entwickeln
- Teambasierte Umgebung mit Versionssteuerung entwickeln

Ziel 2: Webfähige Front-Ends für iSeries-Geschäftslogik erstellen

- Java-Servlets und Java Server Pages (JSP-Dateien) entwickeln, die auf iSeries-Programme und -Daten zugreifen, unabhängig davon, ob Ihre Kernkompetenzen in der Java- oder ILE- und der Original Program Model-Programmierung (Nicht-ILE) liegen
- Darstellung der statischen HTML-Seiten und JSP-Dateien für eine Website anpassen
- Webanwendungen in einer lokalen WebSphere-Testumgebung testen
- Webschnittstelle schnell einer vorhandenen Anwendung hinzufügen und auf einem iSeries-Anwendungsserver implementieren

Ziel 3: GUI-Front-Ends für iSeries-Geschäftslogik erstellen

- Grafische Benutzerschnittstellen für RPG-Programme entwickeln, und zwar unter Verwendung von Tools für die visuelle Entwicklung
- Java-Applets, Java-Anwendungen und native Windows-Anwendungen aus derselben Quelle erstellen

Weitere Entwicklungsziele können mit Hilfe der Tools verwirklicht werden, die in beiden Ausgaben des Development Studio Client enthalten sind, beispielsweise

- XML-Ressourcen entwickeln
- SQL-Anweisungen entwickeln
- Web-Services erstellen
- Webseiten aus lokalen und fernen relationalen Datenbanken generieren

Zugehörige Konzepte

„Paketierung für iSeries-WebSphere-Produkte“ auf Seite 6
„Auswahlmöglichkeiten der Webtechnologie“ auf Seite 13
„iSeries-Entwicklungroadmap“ auf Seite 11
Übersicht über Webanwendungen

Zugehörige Tasks

Verbindung zu einem iSeries-Server herstellen
iSeries-Webanwendungen entwickeln
iSeries-WebFacing-Anwendungen entwickeln

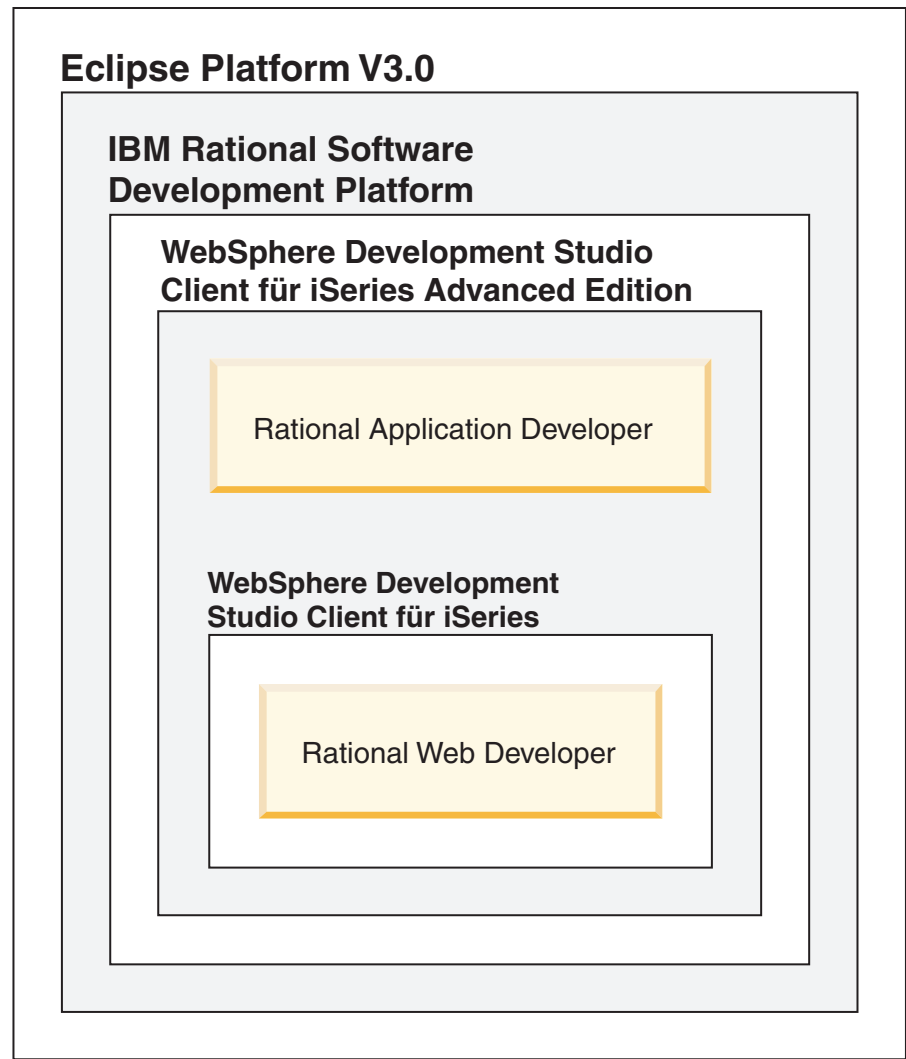
Paketierung für iSeries-WebSphere-Produkte

Zusammenwirken von WebSphere- und Rational-Produkten

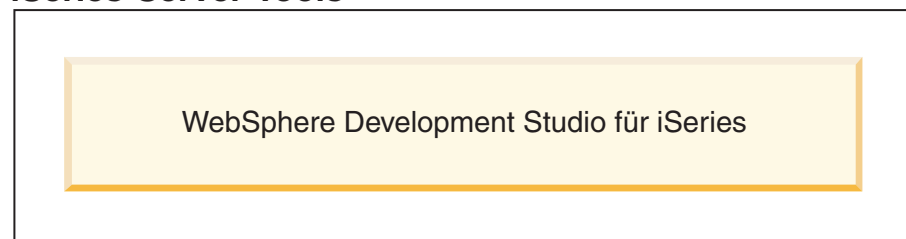
Development Studio Client für iSeries ist zentraler Bestandteil des Angebots mit dem Toolset für die WebSphere-Desktopentwicklung. Die Advanced Edition erweitert dieses Basisprodukt um zusätzliche Funktionen. Analog stellt Rational Web Developer das Basisentwicklungsprodukt dar, das von Rational Application Developer um hoch entwickelte Funktionsmerkmale erweitert wird. Deshalb sind die Advanced Edition und der Rational Application Developer Supersets anderer Produkte.

Wie interagieren die Produktbereiche miteinander? WebSphere Development Studio Client für iSeries baut auf Rational Web Developer auf und fügt eine Reihe von iSeries-Erweiterungen hinzu. Analog basiert die Advanced Edition auf Rational Application Developer und enthält somit alle Funktionen dieses Produkts plus zusätzliche iSeries-Erweiterungen. Insgesamt gesehen, stellt die Advanced Edition ein Superset all dieser Produkte dar; sie umfasst sowohl den Rational Application Developer als auch den Development Studio Client, der wiederum den Rational Web Developer enthält.

iSeries-Client-Tools



iSeries-Server-Tools



Paketinhalt

Bei Bestellung von WebSphere Development Studio für iSeries (das iSeries-Serverprogramm) erhalten Sie auch den WebSphere Development Studio Client für iSeries (das Clientprogramm). Die Server- und Clientprogramme schließen viele Komponenten ein, deren Inhalt nachfolgend genauer erläutert wird.

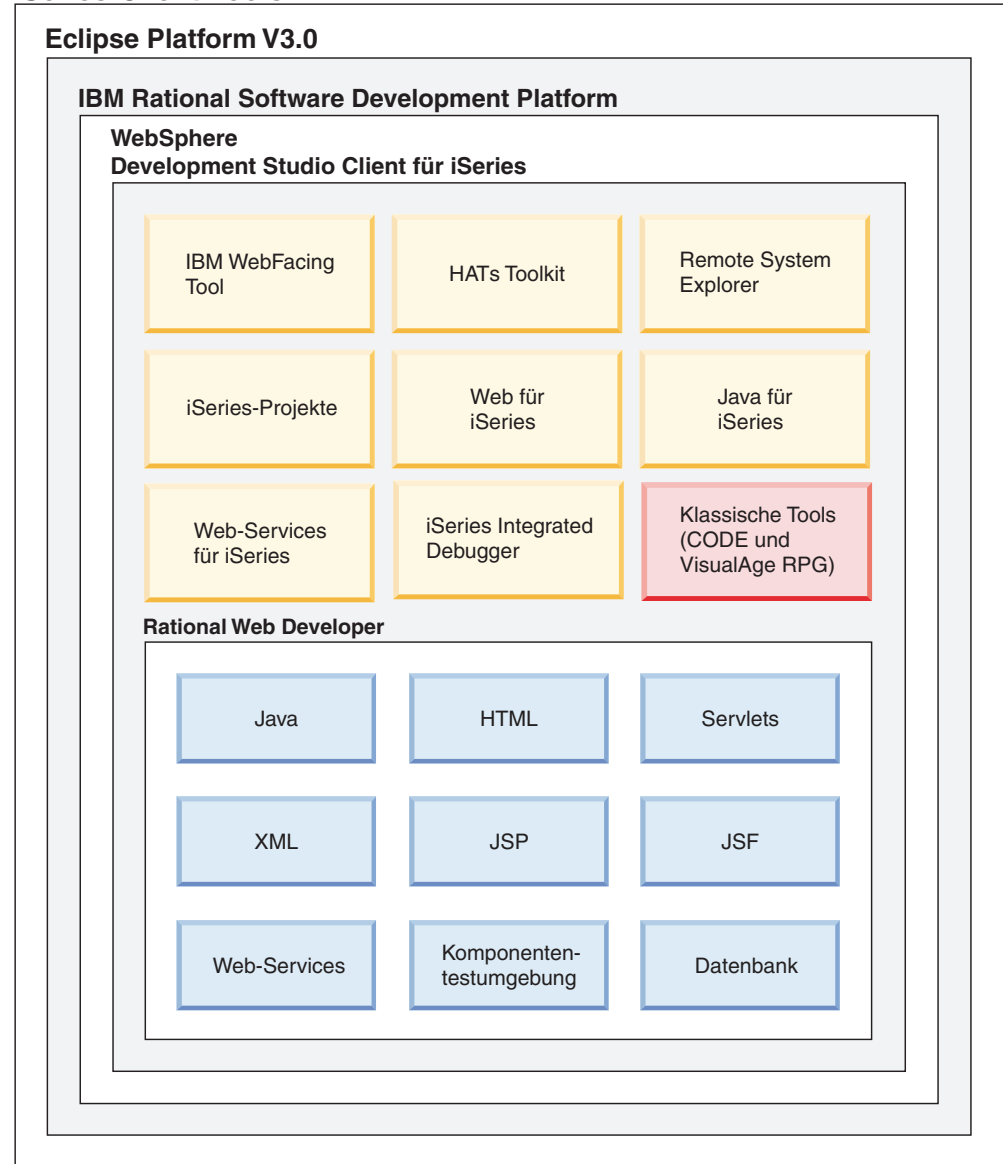
iSeries-Server-Tools

Auf den meisten iSeries-Entwicklungssystemen ist ein Exemplar von WebSphere Development Studio (5722-WDS) installiert. Dieses schließt die Compiler für ILE RPG, COBOL, C und C++ sowie die Serverentwicklungstools wie beispielsweise PDM, SEU und SDA mit ein.

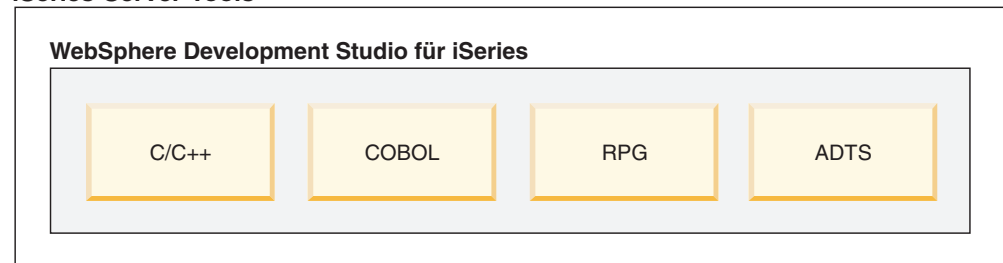
iSeries-Client-Tools

Sie können den WebSphere Development Studio Client für iSeries oder die Advanced Edition auf Ihrer lokalen Workstation installieren und ausführen. Die Eclipse-basierte, integrierte Entwicklungsumgebung (IDE) in beiden Produkten unterstützt weiterhin das wichtige Feld der Entwicklung und Verwaltung traditioneller iSeries-Anwendungen und stellt gleichzeitig aber auch integrierte Tools bereit, mit denen Sie solche Anwendungen ohne großen Aufwand modernisieren können. Die Advanced Edition bietet zwar ein größeres Leistungsspektrum, jedoch nutzen beide Produkte die folgenden Basis- und Paketierungsebenen:

iSeries-Client-Tools

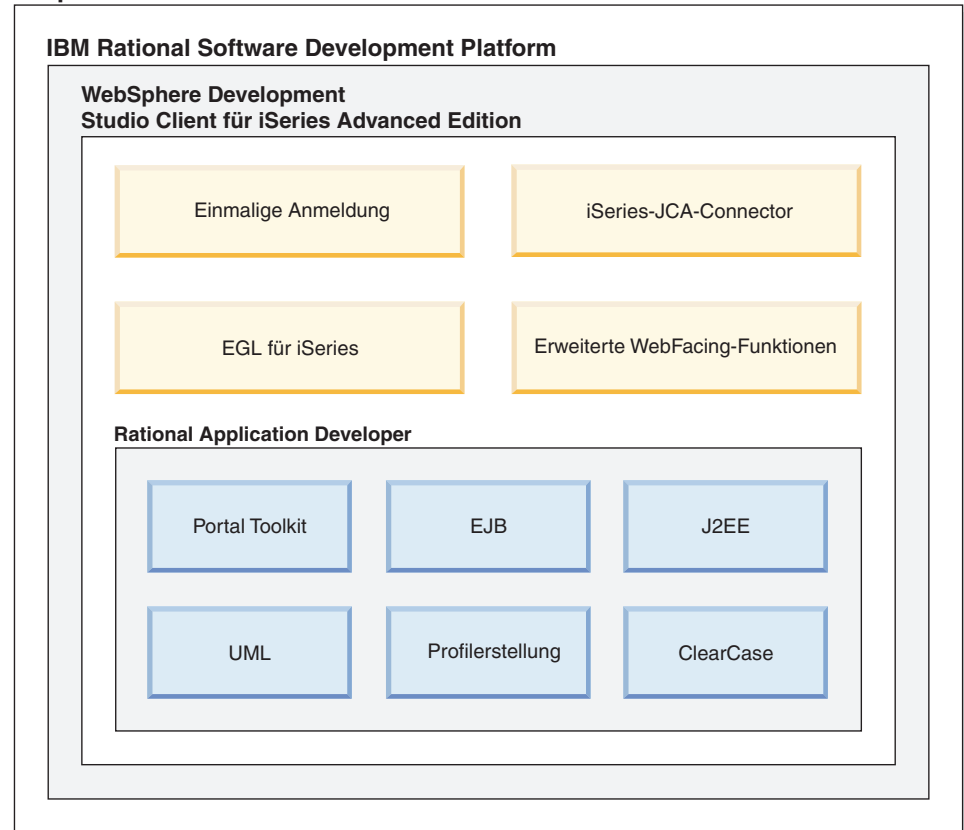


iSeries-Server-Tools



1. **Die Open-Source-Plattform Eclipse 3.0**, mit Common Public License und gebührenfreier weltweiter Verteilung, stellt eine Plug-in-Architektur bereit, die die nahtlose Einbindung in andere Eclipse-basierte Angebote von IBM, externen Geschäftspartnern, ISVs und der Open-Source-Community ermöglicht.
2. **Die IBM Rational Software Development Platform (RSDP)** ist die zentrale, integrierte Entwicklungsumgebung (IDE) und bildet die Basis der zahlreichen Rational- und WebSphere-Produkte, wie beispielsweise Rational Application Developer und beide Development Studio Client für iSeries-Editionen. Wenn Sie eines oder mehrere dieser Produkte installieren, wird die RSDP nur einmal installiert; jedes Produkt wird nahtlos integriert, so dass alle Ihre Tools in einer einzigen IDE zusammenarbeiten. Die RSDP bietet koordinierte Installation und Wartung, konsistente Hilfenavigation, die Perspektive "Willkommen" und rollenbasierte Entwicklung. Diese teamverbindende integrierte Umgebung hat eine konsistente Benutzeroberfläche, die als Workbench bezeichnet wird und der jedes Produkt seine Funktionen hinzufügt.
3. **Rational Web Developer**, der auf RSDP basiert, stellt eine einfach zu handhabende IDE bereit, mit der Sie Web-Services sowie Web- und Java-Anwendungen erstellen, testen und implementieren können. Sie können Drag-and-drop-Anwendungen mit JavaServer Faces (JSF) und Service Data Objects (SDO) entwickeln.
4. **WebSphere Development Studio Client für iSeries** ist das Toolset, das Tools für die Anwendungsentwicklung speziell für iSeries-Entwickler bereitstellt. Es basiert auf Rational Web Developer, so dass bei der Installation von Development Studio Client gleichzeitig auch das Produkt Rational und die RSDP installiert werden. Zusätzlich zu den iSeries-Tools haben Sie über die einheitliche IDE auch Zugriff auf alle Rational Web Developer-Komponenten.
Development Studio Client ersetzt das bisher verwendete Application Development ToolSet (ADTS) als iSeries-Standard-Toolset für Entwickler. Seine zentralen Bestandteile sind Remote System Explorer, das IBM WebFacing Tool und die iSeries-Java- und Web Development Tools. Die klassischen iSeries-Entwicklungstools (CODE und VisualAge RPG) sind zwar nicht mehr Bestandteil der Hauptinstallation, sie sind jedoch als separat installierbare CDs enthalten.
5. **Development Studio Client für iSeries Advanced Edition** bietet alle im Basis-iSeries-Client enthaltenen Funktionen plus zahlreiche weitere Features, wie beispielsweise Unterstützung für einmalige Anmeldung, erweitertes WebFacing-Leistungsspektrum, EGL für iSeries sowie Protokoll- und Traceanalyse. Die Advanced Edition basiert auf Rational Application Developer, so dass Ihnen hoch entwickelte Anwendungsentwicklungstools für Portal-, Java- und J2EE-Anwendungen (J2EE - Java2 Platform Enterprise Edition) zur Verfügung stehen.
Nach Bestellung und Installation der Server- und Clientprogramme müssen Sie sicherstellen, dass die empfohlenen Hardware- und Softwarevoraussetzung für die Workstation erfüllt und die aktuellen iSeries-Server-PTFs und vorläufigen Fixes für Development Studio Client berücksichtigt sind.

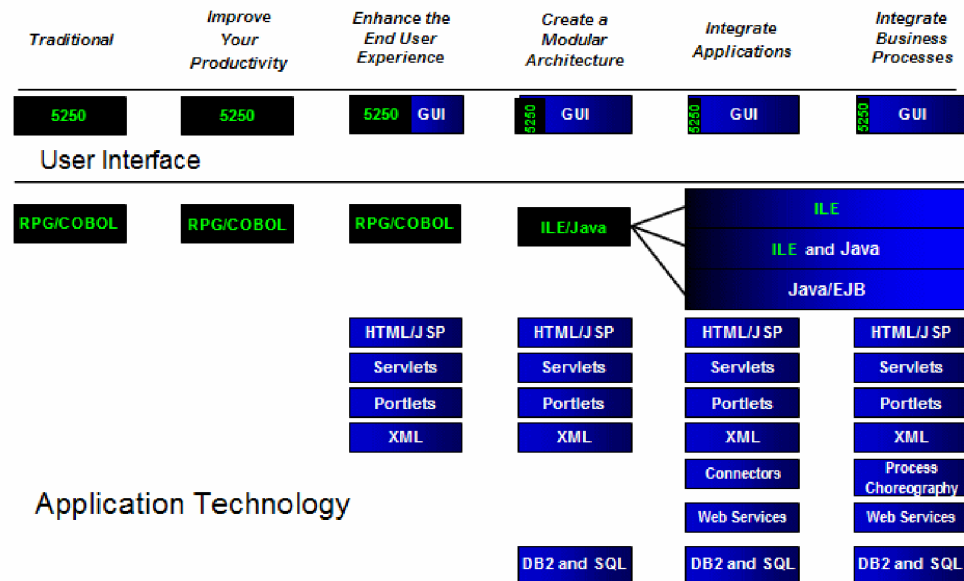
Eclipse Platform V3.0



iSeries-Entwicklungsroadmap

Die iSeries-Entwicklungsroadmap zeigt einzelne Schritte auf, die Entwicklern und Anwendungen den Weg in eine erstklassig implementierte, modernisierte Zukunft weisen. Sie können verschiedene Ansätze verfolgen und unterschiedliche Pakete verwenden, um Ihren Geschäftsanforderungen und branchenspezifischen Erfordernissen gerecht zu werden.

IBM iSeries Developer Roadmap



© 2005 IBM Corporation

iSeries. mySeries.

Die meisten IT-Organisationen und Business Partners, die die iSeries-Plattform verwenden, befinden sich heute auf der linken Seite der Roadmap. In der Regel umfassen die alltäglichen Entwicklungstasks immer noch das Erstellen und Verwalten von Green-Screen-Anwendungen mit Hilfe von traditionellen 5250-Tools, wie z. B. Programmentwicklungsverwaltung (PDM), Quelleneingabedienstprogramm (SEU) und Bildschirmtextentwurfsprogramm (SDA).

Der erste Schritt in Richtung Modernisierung ("Produktivität verbessern") beinhaltet die Verwendung integrierter Entwicklungsumgebungen mit Diagrammen, Desktopfunktionalität und integriertem Testen und Debugging. In diesem Schritt wird Entwicklern die Möglichkeit gegeben, sich auf Geschäftslogik, neue Funktionalität und das Einbeziehen neuer Technologien zu konzentrieren, um dieselbe Entwicklungsarbeit zu leisten, die früher mit PDM, SEU und SDA bewerkstelligt wurde.

Der nächste Schritt, der von Endbenutzern dringend gefordert wird, hat zum Ziel, die Attraktivität für den Endbenutzer zu erhöhen und die mehrere Generationen alten Green-Screen-Anwendungen durch eine verbesserte Benutzeroberfläche zu ersetzen. Für die meisten Anwendungen wird dieser Anforderung am besten durch Umstellung auf browserbasierte Schnittstellen Rechnung getragen; für andere Anwendungen könnte die Client/Server-Architektur oder sogar ein mobiles Endgerät, das zeitnahe und standortunabhängige Technologien verwendet, die Lösung darstellen.

Der Schritt "Modulare Architektur erstellen" ist ein ganz bedeutsamer Schritt, der beinhaltet, dass Sie eine Webanwendung (entweder eine ganz neue Webanwendung oder eine mit Ausschneiden und Einfügen erstellte Webanwendung), eine Anwendungserweiterung oder sogar einen neuen Web-Service erstellen können. Die Trennung der Geschäftslogik von der Benutzerschnittstelle ist eine sehr wichtige Erfahrung und bildet die Grundlage für eine bessere Architektur, die die Wiederverwendung der Logik erlaubt.

Im Schritt "Anwendungen integrieren" geht es darum, neue Technologien einzuführen, um den Zugriff auf Anwendungen, Geschäftslogik und Anwendungen, die Datenbankinformationen integrieren, zu optimieren und dadurch die Wiederverwendung zu maximieren. Beispielsweise wird die Integration von Anwendungen häufig mit Web-Service-Schnittstellen bewerkstelligt, die die Kommunikation zwischen den Anwendungen ermöglichen.

Im letzten Schritt ganz rechts in der Roadmap geht es um die Integration von Geschäftsprozessen. Dabei muss das Zusammenspiel Ihrer Systeme und Anwendungen optimiert und eine Verbindung zu Lieferanten und Kunden hergestellt werden, um Prozesseffektivität, bessere Kommunikation und verbesserten Service zu erreichen.

Weitere Informationen finden Sie auf der Website mit der iSeries-Entwicklungsroadmap <http://www-03.ibm.com/servers/eserver/series/roadmap/>, die auch Links zu Weiterbildungsressourcen enthält.

Auswahlmöglichkeiten der Webtechnologie

Zusätzlich zur iSeries-Entwicklungsroadmap und zu den angesprochenen Neuerungen enthält die folgende Tabelle einen einfachen Leitfaden, der Ihnen die Auswahl der zu verwendenden iSeries-Webtechnologie bzw. des geeigneten Tools unter Berücksichtigung Ihres Programmierzils erleichtern soll. Klicken Sie die Links zu den Referenzinformationen an, um weitere Informationen zu erhalten.

Ihr Ziel	Technologie & Tool	Anmerkungen	Referenzinformationen
Vorhandene interaktive iSeries-Anwendungen webfähig machen	IBM WebFacing Tool Host Access Transformation Services (HATS)	<p>Das IBM WebFacing Tool konvertiert die DDS-Anzeigedateiquelle in JavaServer Pages (JSP-Dateien).</p> <p>HATS wandelt den 5250-Datenstrom zur Ausführungszeit in eine Webbenutzerschnittstelle um.</p> <p>Bei beiden Tools kann die generierte Webbenutzerschnittstelle angepasst werden; darüber hinaus hat jedes der Tools je nach Anforderung andere Stärken.</p>	iSeries-WebFacing-Anwendungen entwickeln

Webbenutzerschnittstellen für iSeries-Geschäftslogik erstellen	Tools für iSeries-Web-Services XML-Toolkit für iSeries	<p>Mit dem Assistenten für Web-Services kann jetzt direkt von der ILE RPG- oder COBOL-Quelle aus ein Web-Service erstellt werden. Alternativ dazu können Sie eine WSDL-Datei (WSDL - Web Service Definition Language) aus der API-Definition eines iSeries-Programms erstellen.</p> <p>Das Toolkit enthält C++-Client-APIs für die Verwendung durch C++-Anwendungen, die Web-Services aufrufen müssen. Eine technische Neuentwicklung stellt eine Reihe von APIs für die Programmiersprache C bereit, die von RPG- und COBOL ILE-Programmen verwendet werden können, um Web-Services aufzurufen.</p>	iSeries-Web-Services entwickeln
Neue Webanwendungen entwickeln	JavaServer Faces (JSF)	Die Technologie 'JavaServer Faces' stellt eine Rahmendefinition und vielseitige Optionen für die Erstellung von Webbenutzerschnittstellen bereit. Es handelt sich um eine strategische Auswahl für die Webanwendungsentwicklung.	Java Server Faces
Rich Client-Anwendungen entwickeln	iSeries Java Development Tools Java Visual Editor	<p>Rich Client-Plattformanwendungen können heutzutage mit den Java-Tools und dem Visual Editor in Development Studio Client erstellt werden. Diesen Eclipse-Workbench-Tools wurden bestimmte iSeries-Leistungsmerkmale hinzugefügt:</p> <ul style="list-style-type: none"> • iSeries-Sicht 'Java umwandeln und ausführen' • Ausführungskonfiguration für iSeries-Java-Anwendung im iSeries-Server • iSeries-Steuerangaben für den Visual Editor, beispielsweise JFormatted-Beans und DFU-Beans 	iSeries-Java-Anwendungen entwickeln

Anwendungen im Hinblick auf zukünftige Technologie-änderungen kapseln und sie für mehrere Plattformen aktivieren	EGL-Tools (Enterprise Generation Language)	EGL ist eine Programmiersprache der vierten Generation (4GL), die Ihre Anwendungen auf einer höheren Ebene zusammenfasst und den entsprechenden Java- oder COBOL-Code auf dem iSeries-Server generiert. Da die Java(TM)- und J2EE-Details verdeckt sind, können COBOL- oder RPG-Programmierer diese Sprache in relativ kurzer Zeit lernen. Sie können dabei ähnliche E/A-Anweisungen verwenden, um auf unterschiedliche Arten von externen Datenspeichern zuzugreifen.	Einführung in EGL
--	--	--	-------------------

Lernprogramme und andere Lernmaterialien

Im Produktumfang sind vielfältige Lernmaterialien enthalten. Wählen Sie die Lernhilfe aus, die für die jeweils auszuführende Task geeignet ist.

Lernprogrammsammlung

Es werden eine Reihe von Lernprogrammen zur Verfügung gestellt, die Ihnen dabei helfen, mit dem Produkt vertraut zu werden. Sie werden in drei Hauptkategorien unterteilt:

- **Watch and Learn (Lernen durch Beobachten):** Mit den Lernprogrammen aus dieser Kategorie können Sie mit Hilfe des enthaltenen Wiedergabegeräts zuschauen, wie eine Task ausgeführt wird, oder ein Tool, das innerhalb des Produkts eingesetzt wird, näher beleuchten.
- **Play and Learn (Lernen durch Üben):** Die Lernprogramme aus dieser Kategorie führen Sie durch eine Simulation, so dass Sie eine Task oder Tätigkeit in der Produktschnittstelle üben können, ohne dabei Ressourcen in Ihrem Arbeitsbereich zu erstellen.
- **Do and Learn (Lernen durch Anwenden)** Die Lernprogramme aus dieser Kategorie richten sich eher an praktisch veranlagte Benutzer mit Mut zum Abenteuer, da diese hier direkt mit dem Produkt interagieren und mit echten Ressourcen arbeiten können.

Alle iSeries-spezifischen Lernprogramme sind in der Kategorie 'Do and Learn' enthalten.

Öffnen Sie die Lernprogrammsammlung durch Auswahl von Hilfe > Lernprogrammsammlung (Tutorials Gallery) im Workbenchhauptmenü.

Beispielsammlung

Die Beispielsammlung enthält alle Beispiele, die Ihnen zur Verfügung stehen. Sie ist in drei Kategorien unterteilt:

- **Showcase-Beispiele** sind die ausführlichsten und leistungsfähigsten Beispiele, die zur Verfügung gestellt werden, und bestehen aus End-to-End-Anwendungen, die den bewährten Verfahren für die Anwendungsentwicklung folgen.
- **Anwendungsbeispiele** werden unter Verwendung von mehr als einem Tool bzw. mehr als einer API erstellt und zeigen, wie die unterschiedlichen Tools interagieren.
- **Technologiebeispiele** sind differenziertere, Code-basierte Beispiele, die sich auf ein einzelnes Tool oder eine einzelne API konzentrieren.

Alle iSeries-spezifischen Beispiele sind in der Kategorie **Technologie** enthalten. Öffnen Sie die Beispielsammlung durch Auswahl von **Hilfe > Beispielsammlung (Samples Gallery)** im Workbenchhauptmenü.

Andere Lernmaterialien

Zusätzliche Lernmaterialien wie beispielsweise Demos, White Papers und Weiterbildungsressourcen finden Sie auf der Webseite für das Produkt, insbesondere auf der Bibliotheksseite bei <http://www.ibm.com/software/awdtools/wdt400/library/index.html>.

Neuerungen in diesem Refresh-Pack

Diesem Refresh-Pack wurden die folgenden neuen Funktionen hinzugefügt. Diese sind nach Task sortiert. Beschreibungen aller Produktkomponenten befinden sich in „Übersicht über Tools“ auf Seite 22.

Task	Neue Funktionen
Allgemeine Entwicklung	<ul style="list-style-type: none"> • IBM WebSphere Development Studio Client V6.0.1 basiert auf dem IBM Rational Web Developer-Refresh-Pack V6.0.1. Rational Web Developer stellt eine IDE (basiert auf Eclipse V3) mit einem breiten Spektrum von grafisch orientierten Tools für die Entwicklung, das Testen und die Implementierung zur Verfügung, die einfach zu verstehen und zu verwenden sind. Rational Web Developer ist Teil der IBM Rational Software Development Platform (RSDP) und umfasst andere Eclipse-basierte Entwicklungstools, mit denen Sie Ihre Entwicklungsumgebungen anpassen und Ihre Entwicklungsumgebungen mit IBM Tools und Partnertools erweitern können. • Updater für Rational-Produkte kann jetzt alle veralteten Dateien auf Ihrem System entfernen, nachdem Sie eine Aktualisierung installiert haben. Sie können die Bereinigungsfunktion verwenden, um den Platten Speicherplatz erheblich zu reduzieren, den das installierte Produkt belegt, und alle veralteten Dateien zu löschen. Zusätzliche Informationen befinden sich unter "Veraltete Dateien entfernen" in der Hilfe zum Updater für Rational-Produkte.
iSeries-Serveranwendungen entwickeln	<ul style="list-style-type: none"> • Mit der Unterstützung für i5/OS V5R4 wurden die V5R4-Sprachänderungen für RPG, COBOL und CL jetzt in die lokalen Syntaxprüffunktionen, Programmprüffunktionen und in die Hilfe integriert. • LPEX-Editor für ferne Systeme bietet jetzt die Möglichkeit, mehrere Sichten derselben Quelle im LPEX-Editor für ferne Systeme für RPG-, COBOL-, DDS-, CL- und CPP-Teildateitypen zu öffnen. • Funktion für automatisches Sichern ist jetzt für den LPEX-Editor für ferne Systeme verfügbar, um Änderungen in dem Editor automatisch in einer lokalen Sicherung zu speichern. • Advanced Verwenden Sie die Protokoll- und Traceanalysetools zum Konvertieren von iSeries-Nachrichten aus Jobprotokollen und Nachrichtenwarteschlangen in das Common Base Event-Format, um die komponentenübergreifende Fehlerbestimmung weiter zu vereinfachen. • WebSphere Development Studio Client Lite V6.0.1-Technologievorschau ist eine einfache Editier-/Kompilier-/Debugumgebung für die Entwicklung von nativen i5/OS-Anwendungen. Sie erstellt eine einfache Workbenchschnittstelle und reduziert den für die Ausführung der Lite-Version erforderlichen Speicherbedarf auf 256 MB. Zum Installieren dieser Zusatzfunktion installieren Sie zuerst die vollständige Standard oder Advanced Version von WebSphere Development Studio Client V6.0.1 und verwenden Sie dann den Updater für Rational-Produkte.

Task	Neue Funktionen
iSeries-WebFacing- und HATS-Anwendungen entwickeln	<ul style="list-style-type: none"> • Webfähige Anwendungen, die mit dem WebFacing Tool erstellt wurden, können jetzt mit Systemanzeigen und DDS-basierten Nicht-WebFacing-Anwendungen interagieren. Für die Produktion erfordert dies die Implementierungsunterstützung, die mit dem IBM WebFacing Deployment Tool für WebSphere Development Studio mit HATS-Technologie zur Verfügung gestellt wird. • HATS Toolkit V6.0.4 enthält eine Standardkonvertierungsengine, mit der schnell und einfach eine dynamische Webschnittstelle aus einem 5250-Datenstrom generiert wird. Die Webschnittstelle kann mit dem Kauf von IBM WebFacing Deployment Tool für WebSphere Development Studio für iSeries - mit HATS-Technologie in der Produktion eingesetzt werden. In diesem Release sind die OLTP-Anforderungen für HATS-Anwendungen in i5/OS V5R4 eliminiert. • Ein neues Beispiel ist eingeschlossen, um diese dynamische Konvertierung des 5250-Datenstroms auf dem WebFacing-Server zu veranschaulichen. • Zusätzliche Unterstützung für BIDI-Sprachen ist für WebFacing-Anwendungen eingeschlossen, wie beispielsweise die Tabulatorsteuerung und die Roundtripfunktion für Webeinstellungen. • Nutzen Sie die Vorteile des Common Base Event-Protokollierungsframeworks für WebFacing- und Web-Tools, um die Fehlerbestimmung in allen Unternehmensanwendungen mit den Protokoll- und Traceanalysetools zu optimieren. • Advanced Die WebFacing-Unterstützung für Cascading Style Sheets-Positioning (CCS-P) ermöglicht es, die Felder in der webfähigen Schnittstelle neu zu positionieren, ungeachtet der Grenzen des ursprünglichen 5250-Bereichs.
Webanwendungen entwickeln	<ul style="list-style-type: none"> • Web-Tools generieren jetzt CBE-Protokolldaten (CBE = Common Base Event), die von den Protokoll- und Traceanalysetools verwendet werden können.
Web-Services entwickeln	<ul style="list-style-type: none"> • Der Assistent für Web-Services unterstützt jetzt die Erstellung eines Web-Service direkt aus ILE RPG- und COBOL-Quelle in der RSE-Perspektive oder aus Projekten in Ihrem Arbeitsbereich. Ähnlich wie bei anderen Web-Service-Typen ist ein Kontextmenüpunkt (Web-Services > Web-Service erstellen) für RPG-, COBOL- und PCML-Quellendateien verfügbar. Der Kontextmenüpunkt ist innerhalb von RSE verfügbar und kann außerdem aufgerufen werden, indem mit der rechten Maustaste auf Dateien im Arbeitsbereich geklickt wird. • Ein neues Beispiel veranschaulicht, wie ein Web-Service aus einem RPG-Serviceprogramm generiert wird.
Debugger für iSeries-Anwendungen ausführen	<ul style="list-style-type: none"> • Ein neues Beispiel zeigt die Verwendung des Protokoll-/Traceanalyseprogramms für Web-Tools und WebFacing-Anwendungen.

Übersicht über Tasks

Dieser Abschnitt beschreibt die Verwendung der verschiedenen Tools im Produkt, mit denen Sie Ihre Hauptentwicklungsziele erreichen können:

- „Ziel 1: Entwicklung und Verwaltung von Anwendungen“ auf Seite 19
- „Ziel 2: Erstellung von Web-Front-Ends“ auf Seite 19
- „Ziel 3: Erstellung von GUI-Front-Ends“ auf Seite 22

Ziel 1: Entwicklung und Verwaltung von Anwendungen

Development Studio Client ist so konzipiert, dass der gesamte Zyklus der Entwicklung und Verwaltung von iSeries-Anwendungen vereinfacht wird. Unabhängig davon, ob Sie mit traditionellen Anwendungen mit grünen Anzeigen arbeiten oder dafür zuständig sind, dass auf bestehende iSeries-Programme als Webanwendungen zugegriffen werden kann, werden Sie feststellen, dass Ihre Produktivität erheblich gesteigert wird, wenn Sie mit den folgenden Tasks arbeiten.

ILE-Anwendungen und Nicht-ILE-Anwendungen editieren und kompilieren sowie Fehler in diesen Anwendungen beheben

Mit Remote System Explorer und iSeries-Projekten können Sie Entwicklungsprojekte auf Ihrem iSeries-Server von Ihrer Windows-basierten Workstation aus erstellen und verwalten. Diese Tools ermöglichen Ihnen die Anzeige von iSeries-Bibliotheken, -Dateien und -Teildateien. Sie können außerdem die Server-Compiler, den Workstationeditor und verschiedene Debugger starten. Ihre Tasks zur Programmeditierung werden mit dem LPEX-Editor für ferne Systeme vereinfacht. Der Editor kann auf Quellendateien auf Ihrer Workstation oder Ihrem iSeries-Server zugreifen.

Mit dem integrierten iSeries-Debugger können Sie Fehler in Ihrem Programm, das auf dem iSeries-Server ausgeführt wird, von einer grafischen Benutzerschnittstelle (GUI) auf Ihrer Workstation aus beheben. Sie können auch Unterbrechungspunkte direkt in Ihrer Quelle definieren, bevor der Debugger ausgeführt wird. Die Clientbenutzerschnittstelle des integrierten iSeries-Debuggers ermöglicht Ihnen zudem, die Programmausführung zu steuern. Sie können beispielsweise Ihr Programm ausführen, Zeilen- und Überwachungsunterbrechungspunkte sowie Unterbrechungspunkte für Serviceeingangspunkte definieren, Programmanweisungen schrittweise durchlaufen sowie Variablen und den Aufrufstack untersuchen. Sie können außerdem von einem einzigen Debuggerfenster aus ein Debug für mehrere Anwendungen ausführen, auch wenn die Anwendungen in unterschiedlichen Sprachen geschrieben sind.

iSeries-Geschäftslogik in Java entwickeln

Mit Hilfe von iSeries Java Development Tools können Sie Java-Anwendungen erstellen, die auf iSeries-Daten zugreifen können. Genauer gesagt, können Sie auch native iSeries-Anwendungen aus Ihrem Java-Code aufrufen.

DDS-Definitionen problemlos erstellen und editieren

Sie können DDS-Definitionen für Anzeigedateien, Druckerdateien und physische Dateien mit Hilfe von CODE Designer erstellen und aktualisieren, und zwar online oder offline. Dieses Tool bietet eine grafische Benutzerschnittstelle, mit der Sie Ihre DDS-Dateien definieren und dann lokal sichern oder direkt auf Ihren iSeries-Server hochladen können. Indem Sie Ihre DDS-Anzeigen visuell erstellen, können Sie deren Benutzerfreundlichkeit verbessern und Ihre Produktivität beim Programmieren steigern (da mehr Zeit für den visuellen Entwurf und weniger Zeit für die Korrektur der Syntax verwendet wird).

Ziel 2: Erstellung von Web-Front-Ends

Mit Development Studio Client können Sie den Zugriff auf Ihre iSeries-Anwendungen über die Green-Screen-Schnittstelle hinaus ermöglichen. Sie können ein webbasiertes Front-End für die DDS-Anzeigen Ihres Programms generieren oder eine neue Webschnittstelle erstellen, die eine Direktverbindung zu den Ein- und Ausgabeparametern Ihres Programms herstellt. Abhängig von Ihrer Programmstruktur, können das IBM WebFacing Tool bzw. die Web Development Tools unter Umständen von größerem Nutzen für Sie sein.

Der zu wählende Weg sollte sich danach richten, wie Ihre iSeries-Anwendung konzipiert ist und welcher Kompromiss zwischen schneller Implementierung und angepasster Darstellung und Funktionsweise getroffen werden soll.

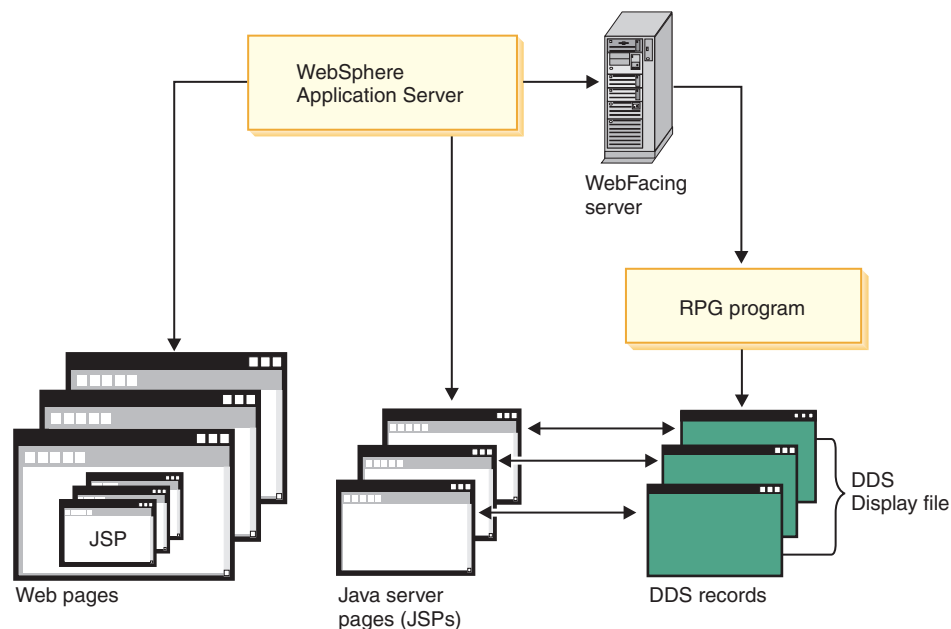
1. Generieren Sie mit dem WebFacing Tool Web-Front-Ends zu bestehenden Anwendungen, bei denen die DDS-Benutzerschnittstelle mit der Geschäftslogik verwoben ist.
2. Erstellen Sie mit Hilfe der Web Development Tools eine neue Webschnittstelle für den Zugriff auf iSeries-Programme und -Daten. Entwickeln Sie eigene Servlets und JSP-Dateien für den Zugriff auf Programme, die für jedes Element der Geschäftslogik über eine Gruppe unterschiedlicher Eingangspunkte und eine Benutzerschnittstellenkomponente verfügen, die in Abhängigkeit von der Benutzereingabe geeignete Aufrufe an Eingangspunkte der Geschäftslogik generiert.
3. Passen Sie die Darstellung der Seiten auf Ihrer Website mit den Web Development Tools an, beispielsweise mit dem Page Designer und den zugehörigen Grafik- und Animationstools WebArt Designer und AnimatedGif Designer.

Web-Front-Ends für bestehende Anwendungen generieren

Mit dem WebFacing Tool können Sie Web-Front-Ends zu iSeries-Anwendungen erstellen, die DDS für ihre Green-Screen-Transaktionen verwenden. Sie wählen eine Webdarstellung aus, generieren eine Gruppe von JSP- und XML-Dateien, die mit der Logik Ihres Programms interagieren, und können dann Ihre Anwendung in der WebSphere-Testumgebung der Workbench testen. Sobald Ihre Anwendung zur Auslieferung bereitsteht, können Sie standardmäßige J2EE WAR- und EAR-Dateien generieren, um sie auf einem WebSphere Application Server zu installieren.

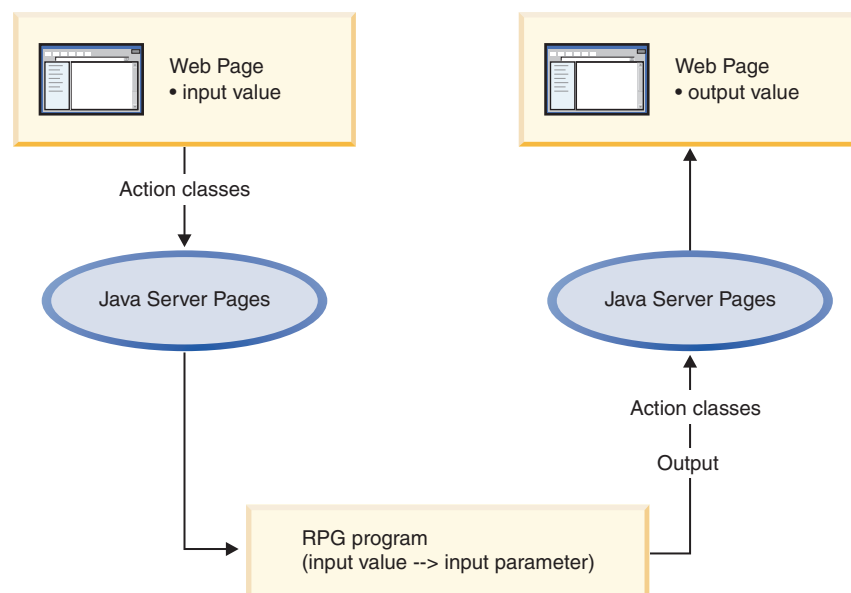
Der Lösungsansatz mit WebFacing ist ideal für Anwendungen, die über ein firmeninternes Netz oder das Internet gesendet werden sollen und dabei die schnelle Bereitstellung Vorrang vor der Anpassung hat. Mit dem WebFacing Tool können Sie weiterhin Ihre Anwendung als 5250-Anwendung zur Verfügung stellen und dieselben ILE-Programme und Nicht-ILE-Programme verwenden, um die Anwendung über das Web zur Verfügung zu stellen.

Die folgende Abbildung zeigt den Laufzeitprozess für das WebFacing Tool:



Neue Webschnittstelle für den Zugriff auf iSeries-Programme und -Daten erstellen

Mit Hilfe der Web Development Tools können Sie auch interaktive Webseiten erstellen. Mit dem Assistenten für Webinteraktion können Sie auf einfache Weise definieren, wie Ihre Seiten mit einer oder mehreren ILE- bzw. Nicht-ILE-Anwendungen interagieren sollen. Dieser Assistent generiert Java-Aktionsklassen und JSP-Dateien für die Verwendung mit Daten aus HTML-Formularen. Gibt der Endbenutzer Daten in einem Formular ein, werden aus der Eingabe Daten für Ihre Programme, und die Ausgabe aus den Programmen wird für das Web formatiert. Sie können die Logik in unterschiedliche Programme für jede Eingabe- und Ausgabephase stellen; dies wird als Webinteraktion bezeichnet. Alternativ dazu kann die Logik auch ein einzelnes Serviceprogramm mit Eingangspunkten für die Handhabung jeder Webinteraktion sein. Die folgende Abbildung zeigt, wie Daten von einem mit den Web Development Tools erstellten Web-Front-End an die iSeries-Anwendung übergeben und an das Web zurückgegeben werden:



Sie können auch Web Development Tools verwenden, um Java-Klassen zu entwickeln, die auf dem iSeries-Server ausgeführt werden. Diese Servlets können iSeries-Daten entweder direkt nutzen, oder sie können über ILE- und Nicht-ILE-Anwendungen auf die Daten zugreifen. Die in den Java Development Tools enthaltenen Klassen der IBM Toolbox for Java erlauben Ihnen, von den Java-Servlets aus auf iSeries-Datendateien und Programmaufrufe zuzugreifen. Sie entwickeln die Servlets mit den Web Development Tools und verwenden die Satz-E/A- und Programmaufrufklassen aus der Toolbox, um iSeries-Daten abzurufen und zu aktualisieren. Sie können Ihre Servlets aus den JSP-Dateien starten, die Sie mit dem Editor von Page Designer erstellen, oder die Ausgabe innerhalb dieser JSP-Dateien anpassen.

Darstellung der Seiten Ihrer Website anpassen

Mit Hilfe der Editiertools in den Web Development Tools können Sie Ihre HTML- und JSP-Dateien anpassen. Außerdem können Sie die Seiten Ihrer Website erweitern und Formularelemente, Java™-Applets, eingebettete Scripts, dynamische Steuerangaben und Tags für JavaServer Pages (JSP) einfügen. Die iSeries-spezifischen Webkomponenten helfen Ihnen bei der Erstellung von Webversionen Ihrer Ein- und Ausgabeseiten mit denselben Typen von Gültigkeitsprüfung, Ausgabe-

formatierung und Subdateisteuerangaben, die auch die nativen DDS-Anzeigen bieten. Für die Ausführung dieser Tasks sind keine fundierten Kenntnisse in HTML oder Java erforderlich.

Ziel 3: Erstellung von GUI-Front-Ends

Sie müssen Ihre iSeries-Anwendungen nicht auf dem Web einsetzen, um ihre Darstellung und Funktionsweise zu modernisieren. Sie können GUI-Front-Ends (GUI - grafische Benutzerschnittstelle) für Ihre iSeries-Anwendungen mit Java Development Tools erstellen.

Applets und native grafische Benutzerschnittstellen aus derselben Quelle erstellen

Nach dem Erstellen einer visuellen Schnittstelle und der zugehörigen Logik auf Ihrer Workstation können Sie Ihre Anwendung als native Windows-Anwendung implementieren oder als Java-Applet, das mit einer geeigneten Java Virtual Machine (JVM™) in jedem Webbrowser ausgeführt werden kann. Dies bietet Ihnen besondere Flexibilität, wenn Sie Ihre Anwendung für Benutzer verfügbar machen. Soll der Zugriff strikt kontrolliert werden, können Sie die Anwendung als ausführbare Windows-Datei einsetzen. Soll der Zugriff nicht derart eingeschränkt werden, können Sie Ihr Applet (und die zugehörigen, von Java Development Tools generierten Dateien) auf eine Website stellen, so dass Benutzer, die über einen Browser auf die Website zugreifen, das Applet in ihrem Browser ausführen und mit Daten auf dem iSeries-Server kommunizieren können.

Java-basierte grafische Benutzerschnittstellen für ILE-Anwendungen und Nicht-ILE-Anwendungen manuell codieren

Sie können Ihre grafischen Benutzerschnittstellen unter Verwendung Ihrer eigenen angepassten Aufrufe für Java-Klassen (z. B. Swing-Klassen) und den Java-Bildschirmmaskeneditor entwickeln und danach die von IBM Toolbox für Java oder JavaBeans (von Java Development Tools zur Verfügung gestellt) bereitgestellten Klassen für den Zugriff auf Ihren iSeries-Server verwenden. Obwohl Ihnen diese Option große Flexibilität im Hinblick auf Darstellung und Funktionsweise Ihres Programms gibt, ist sie dennoch etwas aufwendiger als die Verwendung von Web Development Tools für die Entwicklung von HTML-basierten Schnittstellen für iSeries-Programme.

Übersicht über Tools

Dieser Abschnitt beschreibt die wichtigsten Tools von Development Studio Client und die Arten von Tasks, die mit jedem Tool ausgeführt werden können. Die Tools umfassen:

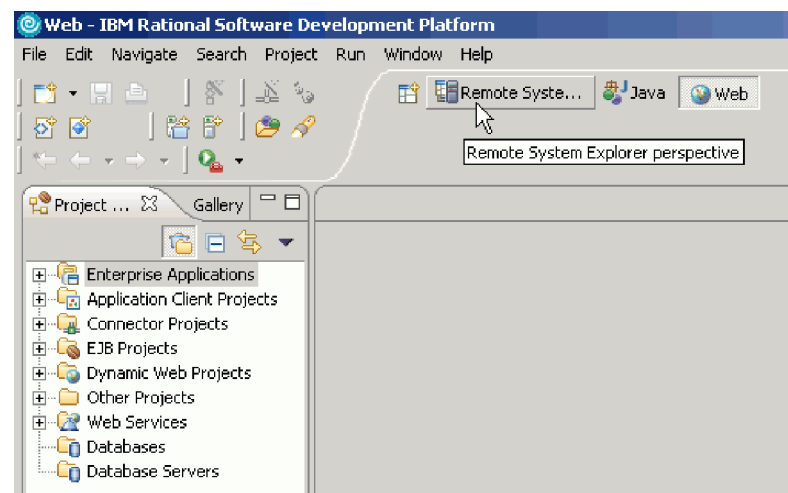
- „Workbench-basierte, integrierte Entwicklungsumgebung“ auf Seite 23
- „iSeries-Entwicklungstools“ auf Seite 23
- „iSeries-Web Development Tools“ auf Seite 27
- „iSeries-Java Development Tools“ auf Seite 28
- „Unterstützung für Struts-Umgebung“ auf Seite 29
- „Entwicklungstools für iSeries-Web-Services“ auf Seite 29
- „Serverentwicklungstools“ auf Seite 30
- „Das IBM WebFacing Tool“ auf Seite 30
- „Integrierter iSeries-Debugger“ auf Seite 31

Workbench-basierte, integrierte Entwicklungsumgebung

Development Studio Client verwendet die Eclipse-Workbench. Diese Workbench bietet Ihnen eine umfassende Entwicklungsumgebung, die alle Ihre Tools, Teams, Ressourcen und anderen Workbench-basierten Projekte nahtlos integriert. Die Plattform stellt das Kerngerüst und die Kernservices zur Verfügung, die als Grundlage für die Erstellung aller Plug-in-Erweiterungen dienen, die Laufzeit, in die Plug-ins geladen und integriert werden und in der Plug-ins ausgeführt werden, sowie ein allgemeines Benutzerschnittstellenmodell für die Arbeit mit e-business Tools. Plug-in-Tools können in die übertragbare die Workbench-API (Anwendungsprogrammierschnittstelle) integriert und dennoch unverändert auf jedem unterstützten Betriebssystem ausgeführt werden. Die Architektur spürt somit diese Plug-ins auf, integriert sie alle in die neue Basisumgebung und stellt somit ein standardmäßiges Benutzernavigationsmodell zum Erstellen integrierter Projekte bereit, die Tools für die Web- und Anwendungsentwicklung enthalten. Da Entwicklungsressourcen für ein Projekt in einem einzigen Repository gespeichert sind, ist die Workbench bestens dafür geeignet, auf gemeinsame Ressourcen zuzugreifen und konsistente Teamunterstützung für gemeinsame Projekte zur Verfügung zu stellen.

Perspektiven

Perspektiven definieren die anfängliche Gruppe und das anfängliche Layout von Sichten in der Workbench. Sie stellen ein Leistungsspektrum bereit, das auf das Ausführen einer bestimmten Art von Task oder das Bearbeiten bestimmter Ressourcentypen abzielt. Um eine Perspektive zu öffnen, klicken Sie **Fenster > Perspektive öffnen** an. Jede geöffnete Perspektive wird durch ein Symbol auf der Symbolleiste oben links im Workbench-Fenster dargestellt. Wenn mehrere Perspektiven geöffnet sind, können Sie durch Anklicken der Symbole auf der Symbolleiste zwischen den Perspektiven hin und herschalten. Ihre aktuelle Perspektive wird in der Titelleiste der Workbench angezeigt.



Es ist auch möglich, angepasste Perspektiven für die Wiederverwendung zu sichern. Um mit Perspektiven zu arbeiten, verwenden Sie die Workbenchmenüpunkte für **Fenster > Perspektive anpassen** und **Fenster > Perspektive sichern unter**.

iSeries-Entwicklungstools

Zwei wichtige iSeries-Entwicklungstools, Remote System Explorer und iSeries-Projekte, stellen das Gerüst, die Benutzerschnittstelle, die Editierfunktion und die ausführbaren Aktionen für iSeries-Objekte, -Befehle und -Jobs zur Verfügung.

Tools für den Zugriff auf und das Editieren von nativen iSeries-Anwendungen

Die iSeries-Entwicklungstools unterstützen Sichten, Editoren und Toolerweiterungen, die Sie beim Entwickeln und Verwalten von iSeries-Anwendungen unterstützen, während Sie mit fernen Betriebssystemen arbeiten. Die Remote System Explorer-Perspektive und die iSeries-Projektperspektive sind die iSeries-Programmierungsumgebungen, die zur Ausführung dieser Tasks verwendet werden. Die Perspektiven führen Sie durch die verschiedenen Verbindungen zu Ihren fernen Systemen, ermöglichen Ihnen den Zugriff auf Dateien und Ressourcen und unterstützen Sie bei der Überwachung von Jobs und der Ausführung von Befehlen. Außerdem stellen sie Unterstützung für die gemeinsame Nutzung der Arbeit über eine Team-Perspektive zur Verfügung. Die meisten Benutzer können ihre Arbeit problemlos über die Remote System Explorer-Perspektive erledigen. Die iSeries-Projektperspektive wurde insbesondere zu dem Zweck entwickelt, strukturiertes Programmieren, Offline-Entwicklung und Teamarbeit zu unterstützen.

Die Remote System Explorer-Perspektive unterstützt auch andere Systemtypen als iSeries-Systeme, wie z. B. UNIX-, Linux- und Windows-Systeme.

Der Remote System Explorer und die iSeries-Projekte stellen folgende zusätzliche Unterstützung zur Verfügung:

- Nahtloser Zugriff auf lokale Dateien und Serverdateien
- Die Möglichkeit zur Verwendung der Unterstützung zum Kopieren, Einfügen und Ziehen und Übergeben, auch zwischen Serversystemen
- Das Bearbeiten von Bibliothekslisten
- PDM-ähnliches Filtern von iSeries-Objekten aus Ihrer Bibliotheksliste
- Die Möglichkeit zur Erweiterung des Remote System Explorer und der iSeries-Projekte durch Verwendung von Tools von Fremdanbietern als Plug-ins
- Eine Tabellensicht, mit der Sie ferne Dateien untersuchen und sortieren, Spaltenanordnungen anpassen, über Funktionstasten PDM-ähnliche Aktionen unterstützen und Befehlszeilenparameter für ferne Objekte angeben können.

Eine gute Stelle für den Beginn im Inhaltsverzeichnis der Hilfetexte ist: Einführung in den Remote System Explorer.

Allgemeine Datei- und Befehlssubsysteme

Sie können Dateien exportieren und importieren, ferne Dateien untersuchen und ferne Befehle für andere Server als iSeries-Server ausführen, wie beispielsweise UNIX-, Linux- und Windows-Server. Bei Verwendung des Remote System Explorer können Sie den Workbench-Editor Ihrer Wahl für jede ferne Datei auf jedem Typ von fernem Server verwenden. Funktionalität für ferne Suche, die das Durchsuchen des Inhalts ferner Dateien und Filter in einem nativen iSeries-Dateisystem erleichtert und asynchrone Suche im iSeries Integrated File System (IFS) sowie auf Linux-, Windows- und UNIX-Servern ausführt.

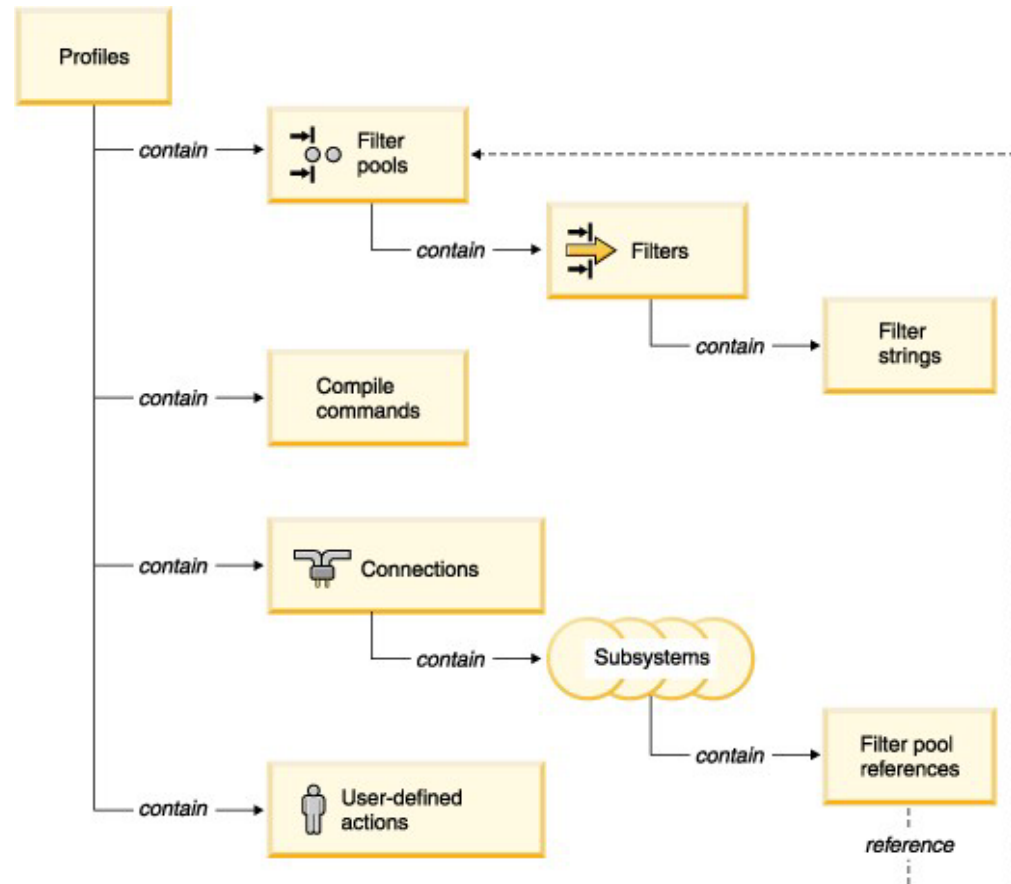
iSeries-Untersuchung und Befehlsausführung

Über den Remote System Explorer können Sie ferne Bibliotheken, Objekte und Quellenteildateien auflisten und diese Elemente mit Hilfe von Filtern verwalten. Sie können Filter für den schnellen Zugriff auf bestimmte Nebenprodukte und Befehle erstellen und gemeinsam benutzen.

Filter können außerdem in Pools untergliedert werden, um die Filter leichter verwalten zu können. Um Zeit zu sparen, können Sie Quellenteildateien auch direkt in der Sicht "Ferne Systeme" öffnen, ohne einen Filter erstellen zu müssen. Sie können mit Dateien des nativen iSeries-Dateisystems (QSYS-Dateien), mit CL-Befehlen

und mit generischen Datei- und Befehlssubsystemen arbeiten, die die Integrated File System-Dateien und QSHELL-Befehle beinhalten. Sie können außerdem für jedes ferne Objekt Ihre eigenen Aktionen und Befehle erstellen.

Die folgende Grafik dient sowohl als Zusammenfassung als auch als Einführung in die Struktur von Remote System Explorer und der iSeries-Entwicklungstools:



iSeries-Aktionen zum Editieren, Kompilieren, Prüfen und Ausführen sowie zum Durchführen einer Fehlerbehebung

Remote System Explorer enthält den LPEX-Editor für ferne Systeme, so dass Sie Quellcode direkt in die Workbench editieren können. Die Schnittstelle stellt Aktionen bereit, die durch Klicken mit der rechten Maustaste aufgerufen werden können, mit denen Sie Ihre Programme kompilieren, ausführen und testen können.

Der LPEX-Editor für ferne Systeme unterstützt viele umfangreiche Editierfunktionen für RPG, COBOL, CL und DDS, wie Quellenbedienerführung, Dateivergleich, Syntaxprüfung, automatische Großschreibung, automatische Einrückung etc.

Auch die folgenden Funktionen helfen Ihnen bei Ihren Programmier tasks:

- Lokale Prüfung und Fehlerrückmeldung
- Die Gliederungssicht als Navigationstool für Ihre Quelle. In der Gliederung können Sie verschiedene Knoten anklicken, um den Cursor an die entsprechende Stelle in Ihrer Datei zu setzen.
- Befehlssshells für die Ausführung lokaler Befehle oder Serverbefehle mit protokollierter Ausgabe

- Die Fähigkeit zum Erstellen einer neuen RPG-Prozedur, einer neuen D-Bestimmung und eines neuen Java-Methodenaufrufs
- Inhaltshilfe (Taste Strg+Leertaste drücken) für die Bereitstellung von Auswahlmöglichkeiten zur Codebeendigung beim Schreiben von Code und automatischer Codebeendigung, wenn es nur eine Übereinstimmung für den Code gibt
- iSeries-Sicht "Jobstatus" für die Bearbeitung von OS/400-Jobs
- Startkonfigurationen ausführen und Startkonfigurationsfehler beheben. Hiermit können Sie Informationen editieren und sichern, die zur Ausführung oder Fehlerbehebung Ihrer Anwendungen erforderlich sind
- Referenzliteratur, die vom Workbenchmenü "Quelle" aus verfügbar ist

iSeries-Projektperspektive

Die iSeries-Projektperspektive stellt das ideale Tool für die iSeries-Anwendungsentwicklung im Team dar. Die Perspektive fördert strukturiertes Programmieren und ermöglicht, dass die Entwicklung ohne Direktverbindung abläuft. In der Perspektive können Sie die Rückmeldung von Fehlern überwachen und Zeitmarken überprüfen, um Konflikte zwischen fernen und lokalen Dateien zu beheben. Zu den funktionalen Erweiterungen für iSeries-Projekte gehört die Funktionalität für die Kompilierung einzelner Teildateien (eine Ergänzung der Funktion, die Ihnen das Erstellen kompletter Projekte ermöglicht).

Erweiterung von Remote System Explorer und iSeries-Projekten für Geschäftspartner

Mit Hilfe der Workbench können Sie Ihre Toolgruppen in den Remote System Explorer integrieren. Viele führende Hersteller von iSeries-Tools integrieren ihre Tools aktiv in das Produkt.

iSeries-Web Development Tools

iSeries Web Development Tools ermöglichen es Ihnen, neue e-business Anwendungen zu erstellen, die ein webbasiertes Front-End verwenden, um mit der Geschäftslogik in einem ILE-Programm oder einem Nicht-ILE-Programm zu kommunizieren, das auf einem iSeries-Server gespeichert ist. Im Gegensatz zum WebFacing Tool können Sie mit Web Development Tools eine ganz neue Anwendung erstellen und das gesamte Anwendungsverhalten und die gesamte Anwendungsdarstellung steuern, wobei Ihnen weitere Tools zur Anpassung zur Verfügung stehen. Sie können Ihre Website auf hohem Niveau entwerfen und Seitenschablonen anwenden, indem Sie den Website Designer verwenden. Anschließend können Sie die einzelnen Seiten mit dem Page Designer erstellen oder JSP-Eingabe- und -Ausgabedateien mit dem Assistenten für Webinteraktion generieren. Sie können Ihren Seiten auch iSeries-Webkomponenten hinzufügen, beispielsweise Webäquivalente der iSeries-Befehlstasten, Eingabefelder, die nur bestimmte Typen von Daten akzeptieren, oder eine Webkomponente, die eine ähnliche Unterstützung wie für Subdateien bereitstellt. Sie können iSeries Web Development Tools sogar verwenden, um die mit dem WebFacing Tool generierten JSP-Dateien anzupassen.

Die Tools bieten Ihnen offene Standards, größere Flexibilität und die Möglichkeit, vorhandene Anwendungen mit Assistenten für Web-Services einzubinden, um Sie beim Schreiben von Anwendungen zu unterstützen, die auf andere Plattformen portierbar sind.

Die spezifischen iSeries-Erweiterungen der Web Development Tools umfassen:

- Unterstützung für die Struts-Laufzeitumgebung und den Struts-Webdiagrammeditor
- Assistent für iSeries-Webtools-Laufzeitkonfiguration, der Informationen zum Servernamen, zur Benutzer-ID, zum Kennwort und zu Bibliothekslisten erfasst
- Der Assistent für Webinteraktion hilft Ihnen bei folgenden Tasks:
 - Sie können die Webkomponenten für eine Webseite mit den Parametern Ihres Programmaufrufs verbinden und die Programmaufrufparameter definieren, ohne sich direkt mit JSP-Dateien und iSeries-Klassen beschäftigen zu müssen.
 - Sie können über eine Webanwendung auf iSeries-Programme zugreifen oder sie als Basis für einen Web-Service einschließen.
 - Zusätzlich zu iSeries-Webkomponenten können Sie JSP-Dateien mit normalen HTML-Steuerelementen verwenden, um Webseiten zu erstellen, die mit einem iSeries-Programm oder Prozeduraufruf interagieren.
 - Sie können eine Seitenschablone für die generierten Eingabe- oder Ausgabeseiten im Assistenten für Webinteraktion auswählen.
 - Mit der Nachrichtenbehandlungsfunktion sind Sie in der Lage, Ihrer Webanwendung im Assistenten für die Laufzeitkonfiguration für iSeries-Webtools eine iSeries-Nachrichtendatei oder eine Java-Merkmaldatei zuzuordnen. Diese Unterstützung ermöglicht es Ihrer Anwendung, Laufzeitnachrichten aus der Merkmal- oder Nachrichtendatei abzurufen und auf Ihren Webseiten anzuzeigen.
 - Die Ablaufcontrollerfunktion hilft Ihnen, mehrere Ausgabeseiten aus dem Webdiagrammeditor miteinander zu verbinden und die Ablaufbedingung im Assistenten für Webinteraktion anzugeben.

- Sie können ein Programm oder eine Prozedur definieren, das bzw. die aufgerufen werden soll, wenn für Ihre Webanwendungssitzung eine Zeitlimit-überschreitung erfolgt.
- Die Flexibilität von Java-Klassen, die mit dem Assistenten für Webinteraktion erstellt wurden, wurde derart erweitert, dass sie jetzt mehr Methoden umfassen und in Paketen zusammengefasst werden können. Diese Java-Klassen werden aus Schablonen generiert, die Sie ebenfalls anpassen können.
- Unterstützung für Cascading Style-Sheets (CSS) – Sie können Ihr eigenes externes oder internes Style-Sheet (oder eigene Inlinedarstellungen) definieren, um die Standarddarstellungen der iSeries-Webkomponenten zu überschreiben. Damit sind Sie beim Anpassen der Darstellung und Funktionsweise (Look-and-Feel) Ihrer Webseiten besonders flexibel.
- Mit der erweiterten Sicht "Palette" ist es einfach, Ihrer eigene JSP-Seite zu entwerfen; außerdem wird in der Sicht "Attribut" eine Gültigkeitsprüfung für Eingabefelder zur Verfügung gestellt, um die Gefahr der Eingabe fehlerhafter Daten zu verringern.

iSeries-Java Development Tools

iSeries Java Development Tools ermöglichen Ihnen, Java-Anwendungen zu entwickeln und Java-Programme zu schreiben, zu kompilieren, zu testen und zu editieren sowie eine Fehlerbehebung für diese Programme auszuführen. Java Development Tools sind ideal zum:

- Entwickeln und Kompilieren der Geschäftslogik in Java, die auf einem fernen Server, beispielsweise einem iSeries- oder Linux-Server ausgeführt wird
- Erstellen von Anwendungen und Java-GUIs, die auf bestehende Serverdaten, Geschäftslogik und Ressourcen zugreifen
- Implementieren oder Exportieren Ihrer Java-Klassen- oder -Quellendateien auf einem bzw. auf einen fernen Server
- Unterstützung für EJB-Abfragesprache, CMP (Container Managed Persistence, über Container realisierte Transaktionspersistenz) und nachrichtengesteuerte JavaBeans
- **Advanced** Verbesserte Verarbeitungsgeschwindigkeit und Speicherleistung für die EJB-Implementierung (EJB = Enterprise JavaBean)

Die spezifischen iSeries-Erweiterungen der Java Development Tools umfassen:

- iSeries-spezifische Unterstützung für Startkonfigurationen, damit Sie Java-Programme auf einem iSeries-Server über Remotezugriff ausführen und eine Fehlerbehebung für Java-Programme durchführen können
- Ein Assistent für Programmaufruf, der JavaBeans und PCML-Dateien (PCML = Program Call Markup Language) erstellen kann, um Ihre iSeries-Programme oder Serviceprogramme aufzurufen. Die JavaBeans können auch verwendet werden, um Web-Services mit Hilfe von Entwicklungstools für Webservices zu erstellen, und Sie können jede mit COBOL- oder RPG-iSeries-Compilern generierte PCML-Datei importieren.
- Eine Sicht "Ferne Fehlerliste" zum Anzeigen der Rückmeldung von Shellbefehlen auf UNIX-, Linux- und Windows-Servern sowie auf lokalen Servern (gültig für Java-/C-/C++-Programmierer)
- Unterstützung für EGL-Programmierung (Enterprise Generation Language) zur Erstellung von Java- oder COBOL-Quelle und zur Vorbereitung der Ausgabe, um ausführbare Objekte zu generieren. Diese Unterstützung ist über die EGL-Perspektive und die EGL-Webperspektive verfügbar.

- Die Fähigkeit, mit der Profildefinitionsperspektive einen Trace für iSeries-Java-Programme durchzuführen und Probleme hinsichtlich der Leistung festzustellen
- Die IBM Toolbox für Java
- iSeries-spezifische JavaBeans
- iSeries-spezifische JVE-Palettenerweiterungen (Java Visual Editor)

Unterstützung für Struts-Umgebung

Development Studio Client bietet Unterstützung für Struts und den Webdiagrammeditor. Struts stellt ein Gerüst zum Erstellen von Webanwendungen dar, und zwar auf Basis des Model-View-Controller-Paradigmas, bei dem die Anwendung in drei separate Komponenten unterteilt wird:

- Model: Das Anwendungsmodell mit entsprechender Datendarstellung und Geschäftslogik
- View: Datendarstellung mit Bereitstellung von Sichten für Benutzereingaben
- Controller: Für die Zuteilung von Anforderungen und die Steuerung des Datenflusses

Der Webdiagrammeditor ist die visuelle Entwicklungsumgebung für das Entwickeln von Struts-basierten Anwendungen. Sie können JSP-Dateien und Aktions-servlets ziehen und an einen unformatierten Bereich übergeben, der ähnlich wie ein leerer Bildschirm ist, Verbindungen zwischen den Elementen herstellen und die diese Elemente dann durch Doppelklicken und Editieren mit Hilfe eines Assistenten definieren.

Entwicklungstools für iSeries-Web-Services

Web-Services sind eigenständige, modulare Anwendungen, die über das Internet beschrieben, publiziert und aufgerufen werden können, was über Just-in-Time-Integration von Services erreicht wird. Sind Komponenten in einem System eng aneinander gekoppelt und basieren sie auf Datenbanksätzen und Flachdateien, sind sie starr und reagieren empfindlich auf Änderungen. Web-Services wirken dem Problem der ineffizienten verteilten Datenverarbeitung entgegen, indem sie sicherstellen, dass Systemkomponenten lose aneinander gekoppelt und dynamisch gebunden werden, um größere Flexibilität, bessere Skalierbarkeit, geringere Systemkosten und daraus folgend größeres Wachstum des Unternehmens zu ermöglichen.

Diese Architektur bringt eine Beziehung zwischen Service Providern, Service Brokern und Serviceanforderern mit sich; Serviceanforderer übergeben eine Anforderung an einen Servicebroker, der wiederum den korrekten Service bei einem Serviceprovider sucht. Webservicesentwicklungstools helfen Ihnen beim Entwickeln solcher Services.

Der Assistent für iSeries-Web-Services erstellt einen Web-Service aus einem iSeries-Stapelprogramm und einer ILE RPG- und COBOL-Quelle, und zwar entweder in der Remote System Explorer-Perspektive oder aus Projekten in Ihrem Arbeitsbereich. Der Assistent für Web-Services arbeitet im Kontext eines Webprojekts und erlaubt das Erstellen, Implementieren, Testen und Generieren eines Proxy sowie die Publizierung in eine UDDI-Registrierungsdatenbank von Web-Services (UDDI = Universal Description, Discovery, and Integration). Diese Services können auf ILE RPG- und COBOL-Quellen, Eingabe- und Ausgabe-URLs, DB2 XML-Erweiterungsaufrufen, gespeicherten DB2-Prozeduren oder auf SQL-Abfragen basieren.

Serverentwicklungstools

Mit Hilfe von Serverentwicklungstools können Sie Ihre Anwendungen in lokal oder fern installierten Laufzeitumgebungen testen. Dazu wird ein Serverprojekt erstellt, das Ihre Umgebung darstellt und Ihre Webprojekte einer Serverkonfiguration zuordnen kann. Somit wissen Ihre Server-Tools, wie das Exemplar für dieses spezifische Webprojekt zu konfigurieren ist.

Serverentwicklungstools umfassen die folgenden Komponenten und Features:

- Die WebSphere-Testumgebung
- Eine lokale Kopie der WAS-Laufzeitumgebung (WebSphere Application Server), so dass Sie Webprojekte und Enterprise Application Archive-Projekte testen können
- Eine lokale Kopie der TCP/IP Monitoring Server-Laufzeitumgebung zum Weiterleiten von Anforderungen und Antworten sowie zum Überwachen der Testaktivität
- Unterstützung für die lokale Apache TomCat-Laufzeitumgebung, so dass Sie Webprojekte testen können, die Servlets und JSP-Dateien enthalten
- Unterstützung für Remote Agent Controller (auf Ihrer fernen Maschine installiert), wenn WAS ein Serverprojektexemplar und eine Konfiguration für Sie erstellen soll, wenn Sie Ihre Webanwendung ausführen (statt dass Sie das Serverprojekt selbst erstellen müssen)

Anmerkung: Verwenden Sie ein externes Serverexemplar, wie beispielsweise WAS auf Ihrer Windows-Maschine oder dem fernen iSeries-Server, müssen Sie Ihr Projekt vor der Ausführung implementieren. Zum Implementieren können Sie Serverentwicklungstools verwenden, um ein fernes Dateiübertragungsexemplar zu Erstellen, das detailliert festlegt, wie und wohin Dateien kopiert werden.

Das IBM WebFacing Tool

Mit dem IBM WebFacing Tool können Sie die Quellenteildateien für Ihre DDS-Anzeigedateien rasch konvertieren, damit die Benutzerschnittstelle Ihres iSeries-Programms in einem Browser ausgeführt werden kann. Beim Konvertieren Ihrer DDS-Anzeigedateien werden JSP- und XML-Dateien für Sie generiert, die den DDS-Code ersetzen und den Webzugriff ermöglichen.

Im Assistenten für WebFacing-Projekte können Sie eine oder mehrere DDS-Quellenteildateien zum Konvertieren auswählen und aus mehreren vordefinierten Darstellungen eine Darstellung und Funktionsweise für das Web auswählen, oder Sie können Ihre eigene Webdarstellung für die Verwendung mit Ihren Anwendungen entwerfen. Das Tool erstellt drei JSP- und XML-Dateien für Ihre Satzformate. Die XML-Dateien enthalten die Daten für das Satzformat oder steuern seine Darstellung oder andere Kenndaten; die JSP handhabt die Anzeige der Webversion am Bildschirm, das Anfordern von Daten und die Bearbeitung von Eingabebefehlen. Der Assistent generiert eine Homepage für die Anwendung, um die webfähige Version Ihres Programms zu starten.

Ruft ein Benutzer eine konvertierte Anwendung über den Browser auf, startet der WebFacing-Server auf dem iSeries-Server die Anwendung. Der Server fängt alle Aufrufe von Operationen READ, WRITE und EXFMT für Anzeigedateien ab, sodass in vielen Fällen Ihr Programm ohne Änderungen ausgeführt werden kann; selbst die Tatsache, dass der Zugriff auf das Programm über das WebFacing Tool erfolgt, wird vom Programm nicht erkannt. Möglicherweise müssen Sie Änderungen am Code vornehmen, wenn Ihre Anwendung DDS-Schlüsselwörter verwendet,

die von WebFacing nicht unterstützt werden, oder wenn die DDS-Anzeigen so geändert werden sollen, dass die Konvertierung in das Webformat ein attraktiveres und konsistenteres Ergebnis liefert.

Das WebFacing Tool stellt außerdem Folgendes zur Verfügung:

- Automatische Konfiguration für die UTF-8-Unterstützung, wenn Sie für WebSphere Application Server implementieren
- Unterstützung für angepasste Tags und Laufzeiterweiterungen
- Unterstützung von Funktionstasten innerhalb von Fenstersätzen
- Unterstützung für Anwendungen, die Felder zwischen Programm und System verwenden, um die Anzeigeattribute für geschützte Felder zu definieren
- **Advanced** Unterstützung für Portletanwendungen, die auf DDS-basierte RPG-, COBOL- und CL-Anwendungen zugreifen können. Sie können die Portletanwendung im Produkt ausführen und testen und die Anwendung dann zu Produktionszwecken in WebSphere Portal Server implementieren.
- **Advanced** Unterstützung für Anwendungen, die während des Anwendungsprozesses Systemanzeigen aufrufen
- **Advanced** Struts-kompatibler Code, der vom Konvertierungsprozess des WebFacing-Tools generiert wird

Integrierter iSeries-Debugger

Mit dem integrierten iSeries-Debugger können Sie Fehler in dem Code, der auf dem iSeries-Server oder auf Ihrem Windows-System ausgeführt wird, über eine grafische Benutzerschnittstelle (GUI) auf Ihrer Workstation beheben. Er unterstützt die folgenden Tasks:

- Verwaltung von Serviceeingangspunkten in Remote System Explorer. Zweck der Serviceeingangspunktfunktion ist es, Ihnen das einfache Debug von Webanwendungen zu ermöglichen, die Geschäftslogik aufrufen, die in RPG, COBOL, CL oder sogar C oder C++ geschrieben ist. Der Serviceeingangspunkt ist eine besondere Art von Eingangsunterbrechungspunkt, der direkt von Remote System Explorer aus definiert werden kann und ausgelöst wird, wenn die erste Zeile einer angegebenen Prozedur in einem Job ausgeführt wird, der sich nicht unter Debugsteuerung befindet. Das heißt, Serviceeingangspunkte ermöglichen es Ihnen, die Steuerung Ihres Job an diesem Punkt zu übernehmen. Eine neue Debugsitzung wird gestartet und die Ausführung an dieser Position gestoppt. Während der ursprüngliche Zweck des Serviceeingangspunkts die Unterstützung beim Ausführen des Debugs für Webanwendungen war, bringt die Verwendung dieser Funktion auch dann Vorteile, wenn das Debug für Anwendungen in einer Umgebung ausgeführt wird, in der der Jobname nicht im Voraus bekannt ist. Einige Beispielszenarios umfassen:
 - Ausführung des Debugs für WebFacing-Anwendungen — In der Regel führt WebSphere Application Server eine JSP-Datei aus, die Ihr RPG- oder COBOL-Programm steuert. Das Debug soll jedoch für den RPG- oder COBOL-Code ausgeführt werden. Da Ihnen der Name des Jobs, in dem Ihr RPG- oder COBOL-Programm ausgeführt wird, nicht im Voraus bekannt ist, ist die Verwendung von Serviceeingangspunkten eine ideale Lösung.
 - Ausführung des Debugs für jede Anwendung, die die Toolbox-Programm- oder -Serviceprogrammaufrufe verwendet — Wird ein Programm oder Serviceprogramm aufgerufen und steht die Ausführung des Codes, in dem der Serviceeingangspunkt definiert ist, an, kann der Debugger die Steuerung der Anwendung übernehmen. Damit können Sie das Programm oder Serviceprogramm, das von der Toolbox aufgerufen wird, unter Debugsteuerung stellen, obwohl Sie nicht wissen, in welchem Job die Ausführung erfolgen wird.

- Wird Ihre Anwendung in einer Mehrjobumgebung ausgeführt, erleichtern Serviceeingangspunkte Ihnen die Ausführung Ihrer Tasks während der Ausführung des Debugs erheblich. Sie können in dem Programm, dessen Ausführung in einem neuen Job ansteht, einen Serviceeingangspunkt definieren. Wird das Programm ausgeführt und steht die Ausführung der Zeile, in der der Serviceeingangspunkt definiert ist, an, kann der Debugger die Steuerung des Programms übernehmen und an dieser Zeile stoppen.
- Definieren von Unterbrechungspunkten in einer Quellencodezeile oder in einer Funktion oder Prozedur und Definieren von Überwachungspunkten, um festzustellen, wann sich eine Variable oder eine Speicherposition ändert
- Drei Arten von Step-Operationen: Step into, Step over und Step return. Sie können auch Step-Operationen für Funktionen oder Prozeduren ausführen, die in unterschiedlichen Programmiersprachen geschrieben sind.
- Anzeigen des Inhalts globaler oder lokaler Variablen, des Aufrufstacks und des Datenspeichers in speziellen Teilfenstern
- Schneller Neustart von Programmen, für die wiederholt ein Debug ausgeführt wird: Unterbrechungspunkte und das Layout von Debuggerfenstern für ein bestimmtes Programm werden automatisch zwischen Debugsitzungen gesichert, sodass sie beim nächsten Mal nicht mehr von Ihnen definiert werden müssen.
- Zuordnung zu einem bereits aktiven Job oder einer bereits aktiven Java Virtual Machine auf dem iSeries-Server.
- Absetzen eines Befehls "Step return" in Ihren ILE-Programmen während der Ausführung des Debugs. Das Absetzen eines Befehls "Step return" erlaubt die Ausführung aller Threads. Der aktuelle Ausführungspunkt wird bis zu der Zeile ausgeführt, die unmittelbar auf die Zeile folgt, die die Funktion oder das Programm aufgerufen hat. Die Ausführung stoppt nach dem Verlassen der aktuellen Funktion oder des aktuellen Programms.
- Ausführung eines Debugs für gespeicherte SQL-Prozeduren, das mit der Anzeige der Editorquellensicht endet.

Der integrierte iSeries-Debugger unterstützt jedes Programm, das in den folgenden Sprachen geschrieben ist:

- ILE RPG, ILE COBOL, ILE CL, C, C++ sowie Original Program Model (Nicht-ILE) RPG, COBOL und CL.
- Java, wenn das Programm unter Windows oder auf einem iSeries-Server ausgeführt wird, auf dem OS/400 V4R2, V5R1, V5R2 oder V5R4 installiert ist. Sie können sogar eine Fehlerbehebung für Servlets, JavaServer PagesTM und Enterprise Java Beans auf dem iSeries-Server ausführen.

Separat installierbare Funktionen

CoOperative Development Environment (CODE) und VisualAge RPG können separat installiert werden.

VisualAge RPG

Mit VisualAge RPG können Sie Client/Server-Anwendungen in einer visuellen Entwicklungsumgebung auf der Workstation entwickeln und verwalten. Sie nutzen Ihre vorhandenen RPG-Kenntnisse, um GUI-Anwendungen (GUI = grafische Benutzerschnittstelle) zu erstellen, die als native Windows-Anwendungen oder als Java-Applets eingesetzt werden können, die auf jedem Java-fähigen Web-Browser ausgeführt werden können. Diese Anwendungen können auf iSeries-Daten und andere iSeries-Objekte zugreifen. VisualAge RPG kann auch eigenständige Java-Anwendungen erstellen.

Mit VisualAge RPG können Sie Anwendungen auf Ihrer Workstation erstellen, editieren und kompilieren sowie Fehler in den Anwendungen beheben. Sie können eine Anwendung von oben nach unten erstellen. Zunächst liegt der Schwerpunkt auf dem Aussehen und der Funktionsweise der Schnittstelle; anschließend werden alle Teile über die in der Sprache VARPG geschriebene RPG-Logik auf der Workstation miteinander verbunden. Die RPG-Logik kann wiederverwendet werden, und Anzeigedateien (DSPF) können aus bestehenden Anwendungen importiert werden.

Dank seiner fest integrierten Tools und seiner visuellen Entwicklungsumgebung können Sie VisualAge RPG mit minimalem Zeitaufwand erlernen. Beispielsweise können Sie rasch Text, Knöpfe und Felder durch Anklicken in Ihrem Entwurfsfenster erstellen.

VisualAge RPG umfasst die folgenden Funktionen und Tools:

- Der VARPG-Compiler schließt die neuesten Änderungen aus dem ILE RPG-Compiler ein.
- Sie können Schnittstellen im GUI Designer erstellen, indem Sie visuelle Komponenten auswählen und an Ihr Entwurfsfenster übergeben, statt Benutzerschnittstellenfähigkeiten (UI-Fähigkeiten) im Quellencode zu codieren. Wählen Sie eine Komponente, wie beispielsweise einen Druckknopf oder ein Eingabefeld, aus, ziehen Sie sie mit der Maus und übergeben Sie sie an das Fenster, das Sie gerade entwerfen. Sie können dann ein Ereignis für die Komponente im Dialogfenstermenü der Komponente auswählen und mit dem Editor die RPG-Programmlogik, die sich hinter dem Ereignis verbirgt, auf der Workstation erstellen. Vom GUI Designer aus können Sie auch den Editor, den Compiler und den Debugger starten.
- Die Syntaxprüfung hilft Ihnen bei der schnellen Erkennung von Syntaxfehlern in Ihren VARPG-Programmen.
- Der Compiler führt Kompilierungen auf der Workstation aus, auf der Ihre Anwendungen schließlich ausgeführt werden.
- Der Hilfe- und Nachrichten-Compiler erlaubt Ihnen die Integration von Onlinehilfetext und Nachrichten in Ihre Workstation-Anwendung.
- Der Editor erlaubt Ihnen das Hinzufügen neuer Editorfunktionen oder das Ändern bestehender Editorfunktionen. Sprachenspezifisches Editieren und sprachenspezifische Hilfe werden zusammen mit dem Editor zur Verfügung gestellt.

CoOperative Development Environment (CODE)

CODE stellt Ihnen eine Reihe von Dienstprogrammen zum Erstellen von Quellen- und DDS-Dateien und zur Verwaltung Ihrer CODE-Projekte zur Verfügung. CODE besteht aus dem CODE Designer und dem CODE-Editor.

CODE Designer

CODE Designer erleichtert die Verwaltung Ihrer DDS-Dateien erheblich. Unter Verwendung eines angepassten Bildschirmmaskeneditors können Sie mit CODE Designer die folgenden DDS-Editier-Tasks ausführen:

- Erstellen neuer DDS-Anzeigen, Druckerdateien und physischer Dateien
- Hinzufügen von Text zu DDS-Anzeigen durch einfaches Anklicken und Eingeben
- Einfügen neuer Anzeigefelder für Ein- und/oder Ausgabe von einem Kontextmenü aus
- Bearbeiten von Feldern und Text mit Funktionen zum Ziehen und Übergeben

- Ändern der Attribute und Merkmale von Datenfeldern und Text
- Verbinden von DDS-Feldern mit iSeries-Datenbankfeldern, indem die Verbindung zu der Datenbank hergestellt wird und die entsprechenden Felder aus einer Pulldown-Liste ausgewählt werden
- Anzeigen der hierarchischen Beziehungen zwischen Dateien, Sätzen, Feldern, Hilfespezifikationen, Schlüsseln und Schlüsselwörtern in jedem ausgewählten DDS-Objekt
- Zusammenfassen von Sätzen in Gruppen für eine bestimmte Anzeige, einen bestimmten Bericht oder eine bestimmte Druckerdatei

Sie können die DDS-Quelle für jedes Element anzeigen, wenn Sie es hinzufügen oder aktualisieren. Sie können sogar den DDS-Code editieren, den CODE Designer für Sie generiert. Sie können außerdem zeitsparender arbeiten, da CODE Designer Fehler markiert, bevor die DDS-Quelle auf dem iSeries-Server kompiliert wird.

CODE-Editor

Der CODE-Editor ist der Vorgänger der Gruppe leistungsstarker Editierfunktionen in der Workbench, in erster Linie im LPEX-Editor für ferne Systeme von Remote System Explorer. Unter „iSeries-Entwicklungstools“ auf Seite 23 befinden sich weitere Informationen.

Kapitel 3. Vorbedingungen

Bevor Sie Webanwendungen unter Verwendung von Development Studio Client und anderen relevanten Tools implementieren, müssen Sie sicherstellen, dass IBM WebSphere Application Server für iSeries und andere Services auf Ihrem iSeries-System korrekt konfiguriert sind. Dieses Kapitel beschreibt die Schritte, die erforderlich sind, um Ihr iSeries-System betriebsbereit zu machen, und stellt Links zu der entsprechenden Dokumentation zur Verfügung. In der Readme-Datei des Produkts und in den Installationshinweisen finden Sie weitere Informationen zu empfohlenen Haupt- und Plattenspeicherplatzzuordnungen.

Planen Sie, Entwicklungsaufgaben mit Remote System Explorer, Java Development Tools oder anderen ILE-bezogenen Tools auszuführen, müssen Sie nur wissen, wie der Zugriff auf Ihr iSeries-System erfolgt, um mit der Entwicklung beginnen zu können; den größten Teil dieses Kapitels können Sie überspringen.

Wenn Sie keine Webanwendungen implementieren müssen, sollten Sie daher an dieser Stelle beginnen: Verbindung zu einem iSeries-Server herstellen.

Mit Ihrem HTTP-Server und WebSphere Application Server arbeiten

Ihre webfähigen iSeries-Anwendungen verwenden WebSphere Application Server, um die Java-Servlets und JavaServer PagesTM (JSPs) auszuführen, die die Kommunikation zwischen dem Browser des Webbenutzers und den iSeries-Programmen oder -Daten erlauben. **Advanced** In Development Studio Client Advanced Edition kann WebSphere Application Server auch Ihre Enterprise Java Beans (EJB) auf Basis der J2EE-Technologie ausführen.

Damit Ihre HTML-Seiten und JSPs dasselbe iSeries-System als Quelle verwenden, müssen Sie auf dem betreffenden Host auch über einen HTTP-Server verfügen. Wir empfehlen die Verwendung des von Apache angetriebenen IBM HTTP-Servers. Die Dokumentation zur Verwendung dieses Servers finden Sie unter: IBM HTTP-Server für iSeries Documentation Center.

WebSphere Application Server handhabt die Ausführung der JavaServer Pages, JavaBeans, Java-Servlets und EJB für Development Studio Client-Prozesse. Die primären Dokumentationsressourcen für IBM WebSphere Application Server für iSeries sind auf den folgenden Websites verfügbar:

- IBM WebSphere Application Server Version 4.0 Advanced Edition für iSeries unter <http://publib.boulder.ibm.com/was400/40/AE/english/docs/>
- IBM WebSphere Application Server Version 4.0 Advanced Single Server Edition für iSeries unter <http://publib.boulder.ibm.com/was400/40/AEs/english/docs/>

Für Versionen 4.0 von WAS müssen als Mindestvoraussetzung die Schritte unter dem Link *Installation* ausgeführt werden. Es wird unbedingt empfohlen, sich mit der Dokumentation zu IBM WebSphere Application Server vertraut zu machen, insbesondere mit den Abschnitten *Installation* und *Mehrere Exemplare des WebSphere Administrative Server definieren*.

Für WebSphere Application Server Version 5.0 und WebSphere Application Server Express:

- RedBooks-Seite für WebSphere: Suchen Sie nach RedBooks für WebSphere Application Server Version 5.0 und WebSphere Application Server Version 5.0 Express Edition unter <http://publib-b.boulder.ibm.com/redbooks.nsf/portals/WebSphere>

Um Ihr System für die Webanwendungsentwicklung und -implementierung vorzubereiten, müssen Sie die folgenden Tasks ausführen. iSeries-spezifische Informationen sind unten berücksichtigt; folgen Sie jedoch den oben erwähnten Links, um Informationen zum IBM HTTP-Server und zu Servern von WebSphere Application Server zu erhalten.

- Mit den Server-Portnummern vertraut machen (siehe nachfolgende Informationen)
- Port für Ihr HTTP-Exemplar suchen
- Port für Ihr WebSphere Application Server-Exemplar suchen
- HTTP-Verwaltungsserverjob starten
- Eigene HTTP-Konfiguration erstellen
- Eigenes HTTP-Exemplar erstellen (optional; Sie können das Standardexemplar verwenden)
- Eigenes HTTP-Exemplar starten
- Beispiellbibliotheken installieren (Informationen werden nachfolgend bereitgestellt)
- Eigenen WebSphere Application Server konfigurieren
- Eigenes WebSphere Application Server-Exemplar starten (Informationen werden nachfolgend bereitgestellt)
- Netzlaufwerk dem iSeries-System zuordnen (Informationen werden nachfolgend bereitgestellt)
- WebSphere-Administrationskonsole starten

Server-Portnummern

Planen Sie, die Standardportnummern für WebSphere Application Server und den HTTP-Server zu verwenden, lauten diese Standardports wie folgt:

- Das HTTP-Serverexemplar verwendet Port 80.
- Das Exemplar von WebSphere Application Server Version 4.0 verwendet Port 900; dieser Port kann jedoch abhängig von der verwendeten Version von WebSphere Application Server abweichen. Für WAS Version 5.0 sind viele Standardports vorhanden. Beispielsweise ist der Verwaltungsport 9090 und der standardmäßige interne HTTP-Port 9080.

Sind dies nicht die korrekten Standardportnummern, bestimmen Sie zusammen mit Ihrem Systemadministrator diese Ports. Sie können diese Portnummern für Ihre HTTP- und WAS-Exemplare verwenden oder Sie können eigene erstellen. Die entsprechende Dokumentation enthält Informationen zum Erstellen Ihrer eigenen Exemplare.

Die folgende Tabelle listet die Ports auf, die Development Studio Client zur Kommunikation mit dem iSeries-Server verwendet:

Development Studio Client-Funktion	Erforderlicher iSeries-Port unter OS/400
Remote System Explorer-Funktionen: einschließlich Filtern, Editieren, Kompilieren und Ausführen	<ul style="list-style-type: none"> • 8470: zentraler OS/400-Server • 8475: ferner OS/400-Befehlsserver • 8476: OS/400-Anmeldeserver • 446: OS/400-DRDA-Server (Satzzugriffs-server) • 449: OS/400-Server-Mapper-Server
Integrated File System-Zugriff (IFS-Zugriff)	8473: OS/400-Dateiserver
Unterstützung für interaktive Jobs (über den Befehl STRRSESVR)	8472: OS/400-Datenwarteschlangenserver
WebFacing (nur für Laufzeit)	4004: WebFacing-Server

Die Remote System Explorer-Ports (8470, 8475, 8476, 446 und 447) müssen für die korrekte Funktion des Produkts offen bleiben. Die Ports für IFS, die Unterstützung für interaktive Jobs und WebFacing müssen jedoch nur offen bleiben, wenn Sie diese Funktionen benötigen.

Gehen Sie wie folgt vor, um die lokalen Ports zu ändern, die Ihre Workstation für die Verbindung zum iSeries-Server verwendet:

- Remote System Explorer-Port:
 1. Klicken Sie in der Menüleiste der Workbench auf **Fenster > Einstellungen**.
 2. Erweitern Sie **Ferne Systeme** und klicken Sie auf **Datenfernverarbeitung**, um diese Option auszuwählen.
 3. Sie können die Portnummer im Feld **Portnummer für RSE-DFV-Dämon** ändern.
- Debugger-Port:
 1. Klicken Sie in der Menüleiste der Workbench auf **Fenster > Einstellungen**.
 2. Erweitern Sie **Debug** und klicken Sie auf **Debugdämon**, um diese Option auszuwählen.
 3. Sie können die Portnummer im Feld **Dämonport** ändern.

Außerdem müssen Sie zwei Ports auf Ihrer lokalen Maschine für die TCP/IP-"Call-back"-Kommunikation mit dem iSeries-Server reservieren.

Development Studio Client-Funktion	Erforderlicher lokaler Port
Unterstützung für interaktive Jobs (über den Befehl STRRSESVR)	3001: Remote System Explorer-DFV-Dämon
Integrated Debugger	8001: Debuggerdämon. Diese Funktion verwendet ebenfalls Port 3001.

Beispielbibliotheken installieren

Um mit den Beispielen in diesem Handbuch zu arbeiten, müssen Sie die Bibliotheken WHOLESale und WDSCLAB auf Ihren iSeries-Server zurückspeichern. Sie sollten dies selbst dann ausführen, wenn Sie die Bibliotheken bereits für ein vorheriges Release des Produkts zurückgespeichert haben, da sich ihr Inhalt möglicherweise geändert hat. Statt des 5250-Emulators können Sie die Eclipse-Workbench verwenden, um die Sicherungsdateien für die Bibliothek zurückzuspeichern. Die Anweisungen beschreiben, wie die Bibliothek WHOLESale zurückgespeichert wird. (Die Anweisungen zum Zurückspeichern der Bibliothek WDSCLAB sind identisch, mit der Ausnahme, dass in der Workbench wdsclab.savf statt wholesale.savf ausgewählt werden muss.) Der folgende Prozess lädt die Sicherungsdatei in eine temporäre Bibliothek auf Ihrem iSeries-Server, QRSETEMP, hoch und speichert diese Sicherungsdatei dann in eine neue Bibliothek mit dem Namen *SAVLIB oder eine Bibliothek gemäß Ihrer Angabe zurück.

Anmerkung: Die Sicherungsdateien zum Installieren der Beispielbibliothek sind für die Verwendung eines iSeries-Systems bestimmt, auf dem V5R1 oder höher installiert ist.

Die Bibliotheken werden wie folgt zurückgespeichert:

1. Schalten Sie in der Workbench in die Remote System Explorer-Perspektive um, falls dies noch nicht geschehen ist. Klicken Sie im Menü der Workbench auf **Fenster > Perspektive öffnen > Remote System Explorer**.
2. Erweitern Sie in der Sicht "Ferne Systeme" **Lokal > Lokale Dateien > Laufwerke**.
3. Erweitern Sie das Verzeichnis **Laufwerke**, um zu dem Verzeichnis zu navigieren, in dem Sie das Produkt installiert haben. Standardmäßig ist dies **c:\WDSC**.
4. Erweitern Sie im Verzeichnis **WDSC wdsksampl**.
5. Klicken Sie mit der rechten Maustaste auf **wholesale.savf** und wählen Sie **Auf iSeries zurückschreiben** aus, um das Dialogfenster "iSeries-Sicherungsdatei zurückschreiben" aufzurufen.
6. Das erste Feld, **iSeries-Verbindung**, gibt den iSeries-Server an, auf den die Sicherungsdatei zurückgeschrieben werden soll. In der Workbench verwenden Sie eine Verbindung für die Kommunikation zwischen Ihrer lokalen Workstation und dem iSeries-Server. Ist keine Verbindung vorhanden, klicken Sie auf **Neu**, um eine Verbindung zu einem iSeries-Server zu definieren. Ist bereits eine Verbindung vorhanden, wird in diesem Feld automatisch der Verbindungsname vorgegeben.
7. Lassen Sie die Standardwerte für die folgenden Felder unverändert: **Sicherungsdateibibliothek** und **Sicherungsdateiname**.
8. Geben Sie **WHOLESale** in das Feld **Gesichert aus Bibliothek** ein.
9. Im Feld **Zurückschreiben in Bibliothek** können Sie den Standardwert akzeptieren, den Namen einer vorhandenen Bibliothek eingeben, die verwendet werden soll, oder eine neue Bibliothek erstellen, indem Sie einen neuen Namen, wie z. B. **MYLIB**, eingeben.
10. Klicken Sie auf **OK**. Geben Sie auf Anforderung Ihre iSeries-Benutzer-ID und Ihr Kennwort ein.
11. Kehren Sie nach Beendigung des Assistenten zur Sicht "Ferne Systeme" zurück, und suchen Sie **wdsclab.sav** in demselben Verzeichnis (**Lokal > Lokale Dateien > Laufwerke > C > WDSC > wdsksampl**).

12. Klicken Sie mit der rechten Maustaste auf **wdsclab.sav**, und wählen Sie **Auf iSeries zurückschreiben** aus, um das Dialogfenster "iSeries-Sicherungsdatei zurückschreiben" aufzurufen.
13. Wählen Sie dieselbe iSeries-Verbindung wie zum Zurückschreiben der Sicherungsdatei WHOLESale im Feld **iSeries-Verbindung** aus.
14. Lassen Sie die Standardwerte für die folgenden Felder unverändert: **Sicherungsdateibibliothek** und **Sicherungsdateiname**.
15. Geben Sie WDSCLAB in das Feld **Gesichert aus Bibliothek** ein.
16. Im Feld **Zurückschreiben in Bibliothek** können Sie den Standardwert akzeptieren, den Namen einer vorhandenen Bibliothek eingeben, die verwendet werden soll, oder eine neue Bibliothek erstellen, indem Sie einen neuen Namen, wie z. B. MYLIB2, eingeben.
17. Klicken Sie auf **OK**. Geben Sie auf Anforderung Ihre iSeries-Benutzer-ID und Ihr Kennwort ein.

Beispielbibliothek Ihrer Anfangsbibliotheksliste hinzufügen

Damit Ihre neu hinzugefügte Bibliothek permanent angezeigt wird, wenn Sie Ihre Bibliotheksliste im Remote System Explorer erweitern, können Sie Ihre Jobbeschreibung (ist Ihrem Benutzerprofil zugeordnet) ändern, um die Bibliothek Ihrer Anfangsbibliotheksliste hinzuzufügen. Gehen Sie wie folgt vor, um die Bibliothek hinzuzufügen:

1. Öffnen Sie einen 5250-Emulator und melden Sie sich bei Ihrem iSeries-Server an.
2. Geben Sie `dspusrprf <Benutzer-ID>` ein, um Ihr Benutzerprofil anzuzeigen.
3. Drücken Sie die Taste zum Vorblättern, um durch Ihr Profil zu blättern, bis Sie die Zeilen für die Jobbeschreibung (Job description) und die Bibliothek (Library) finden. (Die Bibliothek steht direkt nach der Jobbeschreibung.)
4. Stellen Sie sicher, dass es sich um Ihre Jobbeschreibung bzw. um die von Ihrem Team verwendete Jobbeschreibung handelt. Ist dies nicht der Fall, erstellen Sie Ihre eigene Jobbeschreibung und ordnen Sie sie Ihrem Benutzerprofil zu. Wenden Sie sich an Ihren Systemadministrator, falls Sie Hilfe benötigen.
5. Kehren Sie zum Hauptmenü zurück und geben Sie `chgjobd` ein. Drücken Sie dann F4.
6. Geben Sie in das Feld für die **Jobbeschreibung** den Namen und die Bibliothek für Ihre Jobbeschreibung ein. Ist Ihnen der Name Ihrer Jobbeschreibung nicht bekannt, führen Sie ein `dspusrprf` aus, um Ihr Benutzerprofil anzuzeigen. Ihre Jobbeschreibung befindet sich auf einer der Seiten Ihres Benutzerprofils.
7. Drücken Sie die Funktionstaste 10, um auf zusätzliche Parameter zugreifen zu können.
8. Drücken Sie die Taste zum Vorblättern, um durch Ihre Jobbeschreibung zu blättern, bis Sie die Zeile Anfangsbibliotheksliste (Initial library list) finden.
9. Geben Sie in das Feld neben der Zeile, die + für weitere Werte angibt, ein + gefolgt von einem Leerzeichen ein und drücken Sie dann die Eingabetaste.
10. Geben Sie in der Anzeige **Weitere Werte angeben** WHOLESale in das erste verfügbare Feld ein und drücken Sie die Eingabetaste, um Ihre Jobbeschreibung zu ändern. Wiederholen Sie diesen Schritt für WDSCLAB.

Eigenes WebSphere Application Server-Exemplar starten

Sie müssen Ihr eigenes WebSphere Application Server-Exemplar jedes Mal erneut starten, wenn es gestoppt wurde, und jedes Mal, wenn Ihr iSeries-System erneut gestartet wird. Führen Sie diese Schritte aus:

1. Melden Sie sich am iSeries-System an und öffnen Sie eine QShell-Eingabeaufforderung, indem Sie den Befehl `qsh` verwenden.
2. Wechseln Sie zu dem Verzeichnis `bin` des Installationsverzeichnisses von WebSphere Application Server, indem Sie den Befehl `cd` verwenden.
3. Geben Sie den folgenden Befehl ein:
`strwasinst -instance wasinst`

Dabei gilt:

wasinst

ist der Name des WebSphere Application Server-Exemplars, das Sie verwenden, und

Beachten Sie, dass Sie für WebSphere Application Server Express Edition den Befehl `startServer` an Stelle des Befehls `strwasinst` verwenden.

4. Ist die Ausführung des Befehls beendet und wird ein Eingabeaufforderungszeichen (ein Dollarzeichen, `$`) zusammen mit einer Bestätigungsnachricht an der Konsole angezeigt, verlassen Sie QShell durch Drücken der Taste F3.

Netzlaufwerk dem iSeries-System zuordnen

Web Development Tools implementieren Ihre HTML-, JSP- und Servlet-Dateien sowie andere Dateien, die publiziert werden können (EJB für Development Studio Client Advanced Edition), auf dem iSeries-System in einem gemeinsamen Netzlaufwerk. Können Sie dem Verzeichnis `/QIBM` auf Ihrem iSeries-System ein Netzlaufwerk mit dem Befehl `NET USE` oder dem Befehl zum Verbinden eines Netzlaufwerks zuordnen, ist der `NET USE`-Zugriff bereits korrekt definiert. Führen Sie diese Schritte aus, um den `NET USE`-Zugriff, abhängig von Ihrem Betriebssystem, für das iSeries-System zu definieren.

Für Windows 2000 und XP:

1. Öffnen Sie eine Eingabeaufforderung und geben Sie den Befehl `NET USE` ein.
2. Wird ein Laufwerksbuchstabe angezeigt, der dem Verzeichnis `/QIBM` auf Ihrem iSeries-System zugeordnet ist, ist der `NET USE`-Zugriff bereits definiert.
3. Geben Sie den folgenden Befehl ein: `NET USE * \\MYHOST\QIBM /USER:USERNAME *`; dabei ist *USERNAME* Ihre iSeries-Benutzer-ID.
4. Geben Sie, wenn aufgefordert, Ihr iSeries-Kennwort ein.

Können Sie erfolgreich die Verbindung zu dem Netzlaufwerk herstellen, ist der `NET USE`-Zugriff korrekt definiert. Wird stattdessen eine Fehlermeldung angezeigt, die auf Probleme beim Herstellen der Verbindung zum iSeries-System hinweist, prüfen Sie zusammen mit dem Systemadministrator, ob TCP/IP auf dem iSeries-System korrekt gestartet wurde. Verfügen Sie über die Administratorberechtigung, können Sie TCP/IP selbst starten, indem Sie sich am iSeries-System anmelden, `STRTCPSVR *NETSVR` eingeben und alle Eingabeaufforderungen wie erforderlich beantworten. Ist die Ausführung dieses Befehls beendet, versuchen Sie erneut, die oben beschriebenen Schritte auszuführen. Können Sie Ihrem iSeries-System kein Netzlaufwerk zuordnen, benachrichtigen Sie den Systemadministrator.

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in dieser Dokumentation beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieser Dokumentation ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Europe Director of Licensing 92066 Paris La Defense Cedex France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

Lab Director IBM Canada Ltd. Laboratory 8200 Warden Avenue Markham, Ontario, Canada L6G 1C7

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in dieser Dokumentation aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

(C) (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. (C) Copyright IBM Corp. 2000, 2005. Alle Rechte vorbehalten.

Informationen zu Programmierschnittstellen

Die Informationen zu den Programmierschnittstellen sollen Ihnen helfen, Anwendungssoftware mit dem vorliegenden Programm zu erstellen.

Allgemeine Programmierschnittstellen ermöglichen Ihnen das Schreiben von Anwendungssoftware, mit der die Services der Tools des vorliegenden Programms abgerufen werden.

Diese Informationen können jedoch außerdem Diagnose-, Änderungs- und Optimierungsinformationen enthalten. Die Diagnose-, Änderungs- und Optimierungsinformationen helfen Ihnen beim Durchführen einer Fehlerbehebung für Ihre Anwendungssoftware.

Warnung: Verwenden Sie diese Diagnose-, Änderungs- und Optimierungsinformationen nicht als Programmierschnittstelle, da sie Änderungen unterliegen.

Marken und Servicemarken

Folgende Namen sind in gewissen Ländern (oder Regionen) Marken der International Business Machines Corporation:

AIX
AIX-Fenster
Application System/400
AS/400
AS/400e
C Set ++
COBOL/2
COBOL/400
DB2
DB2 Extenders
DB2 Universal Database
IBM
IBMLink
Integrated Language Environment
iSeries
Language Environment
MQSeries
Open Class
Operating System/400
OS/2
OS/390
OS/400
RPG/400
SQL/400
VisualAge
WebSphere

InstallShield ist eine Marke der InstallShield Corporation.

Intel und Pentium sind in gewissen Ländern Marken der Intel Corporation.

Java und alle Java-basierten Marken und Logos sind in gewissen Ländern Marken oder eingetragene Marken von Sun Microsystems, Inc.

Rational ist in gewissen Ländern eine Marke der International Business Machines Corporation und der Rational Software Corporation.

ActiveX, Microsoft, SourceSafe, Visual C++, Visual SourceSafe, Windows, Windows NT, Win32, Win32s und das Windows-Logo sind in gewissen Ländern Marken der Microsoft Corporation.

UNIX ist eine eingetragene Marke von The Open Group.

Andere Namen von Unternehmen, Produkten und Services können Marken oder Servicemarken anderer Unternehmen sein.

IBM