

IBM

@server

iSeries

Управление iSeries Access для Windows





@server

iSeries

Управление iSeries Access для Windows

Содержание

Управление iSeries Access для Windows	1
Новое в версии V5R2	2
Как напечатать этот раздел	3
Сетевые среды iSeries Access для Windows	3
Microsoft Windows Terminal Server	4
Применение iSeries Access для Windows в трехуровневой среде	4
Применение Microsoft Transaction Server (MTS)	5
Обеспечение доступа к службам iSeries компонентам среднего уровня.	6
Добавление всех пользователей в конфигурацию TCP/IP	8
Задание расположения файлов PC5250 для всех пользователей	8
Пользовательские профайлы для многопользовательских PC.	8
Установка и обновление программы на нескольких PC	9
Создание специального установочного образа iSeries Access для Windows	10
"Тихая" установка iSeries Access для Windows	11
Создание файлов ответов для установки iSeries Access для Windows	12
Запуск "тихой" установки	13
Коды возврата для "тихой" установки или перехода	13
Работа с пакетами обслуживания	14
Проверка уровня обслуживания	15
"Тихая" установка пакета обслуживания	15
Администрирование ODBC	15
Обзор драйвера ODBC iSeries Access	16
Настройка системы для применения драйвера ODBC iSeries Access	17
Добавление локальной системы в каталог RDB	18
Выбор источника данных ODBC	18
Защита ODBC iSeries Access для Windows.	19
Рискованные стратегии защиты ODBC	19
Стратегии защиты программ ODBC	20
Информация, связанная с защитой ODBC	21
Устранение неполадок ODBC	22
Инструменты диагностики и анализа производительности ODBC	22
Сообщения об ошибках ODBC iSeries Access	23
Устранение неполадок при подключении к серверу iSeries	25
Общие ошибки ODBC	27
Сбор информации для IBM Support	29
Администрирование серверов хоста	30
Серверы хоста OS/400	30
Список серверов хоста iSeries Access для Windows, упорядоченный по их назначению	31
Файловый сервер	32
Сервер баз данных	33
Сервер очередей данных	38
Сервер сетевой печати	38
Центральный сервер	39
Сервер обработки удаленных команд и вызова распределенных программ.	39
Сервер входа в систему	40
Сервер номеров портов	40
Работа с серверами хоста OS/400.	40
Настройка соединений между клиентом и сервером	41
Подсистемы на сервере iSeries.	45
Системные значения сервера iSeries	57
Просмотр списка заданий серверов в системе iSeries	60
Применение программ EZ-Setup и Навигатор iSeries для работы с серверами хоста	63
Работа с программами выхода сервера	63

Регистрация программ выхода	64
Создание программ выхода	67
Параметры программ выхода	67
Примеры: Программы выхода	85
Добавление новых функций в iSeries Access для Windows и Навигатор	99
Работа со встраиваемыми модулями	100
Работа с дополнительными модулями	101
Управление сервером iSeries NetServer	102
Ограничение возможностей пользователей с помощью стратегий и Администрирования приложений	102
Обзор стратегий iSeries Access для Windows	103
Типы стратегий и области их применения	104
Настройка системы для применения стратегий	105
Настройка сервера iSeries для работы со стратегиями	105
Настройка PC для загрузки стратегий	105
Создание файла стратегий	106
Список стратегий iSeries Access для Windows	108
Список стратегий, упорядоченный по их назначению	108
Список стратегий, упорядоченный по шаблонам	111
Администрирование Secure Sockets Layer	111

Управление iSeries Access для Windows

В этом разделе предполагается, что вы уже знакомы с программой iSeries Access для Windows и она установлена в вашей системе. Общий обзор программы iSeries Access для Windows и описание способов ее применения в сети приведены в разделе Начало работы. Инструкции по установке и настройке iSeries Access для Windows приведены в разделе *iSeries Access для Windows -Установка*



В этом разделе приведена информация по управлению продуктом iSeries Access для Windows.

Сетевые среды iSeries Access для Windows

В этом разделе рассматриваются сетевые среды, в которых может работать iSeries Access для Windows. В частности, здесь приведены различные варианты предоставления клиентам доступа к службам OS/400: в трехуровневой среде iSeries Access для Windows или путем установки этого продукта в операционной системе Windows NT Server 4.0 Terminal Server Edition или Windows 2000 со службами терминала. Кроме того, здесь приведена информация об администрировании многопользовательских PC.

Установка или перенос на несколько PC

Для выборочной установки iSeries Access для Windows на нескольких PC не обязательно каждый раз выполнять операции первоначальной установки и настройки.

Администрирование пакетов обслуживания

В этом разделе приведено описание PTF и пакетов обслуживания, а также способов управления ими с помощью функции проверки уровня обслуживания.

Администрирование ODBC

В iSeries Access для Windows входит драйвер ODBC, который представляет собой удобное средство сетевого доступа к базе данных DB2 UDB для iSeries для приложений. В этом разделе приведены описание ODBC, инструкции по настройке драйвера и руководство по устранению неполадок.

Дополнительная информация по реализации и применению API ODBC приведена в разделе Программирование ODBC.

Управление серверами хоста

В этом разделе описаны серверы хоста, наиболее часто применяемые с iSeries Access для Windows, и способы наиболее эффективной работы с ними.

Добавление новых функций в iSeries Access для Windows и Навигатор

Стандартный набор функций iSeries Access для Windows и Навигатора iSeries можно расширить за счет особых приложений, называемых дополнительными (встраиваемыми) модулями. В этом разделе описаны способы добавления встраиваемых модулей и дальнейшей работы с ними в iSeries Access для Windows.

Установка ограничений с помощью стратегий и функции Администрирование приложений

В iSeries Access для Windows предусмотрено много способов ограничения деятельности пользователя. В их число входят стратегии, которые можно задать с помощью редактора стратегий Microsoft, и функция администрирования приложений Навигатора iSeries.

Для эффективного управления iSeries Access для Windows рекомендуется предварительно ознакомиться с рядом разделов. Эти разделы перечислены ниже:

- Secure Sockets Layer (SSL)
- AS/400 NetServer
- iSeries Access for Windows on Windows 2000 implementation notes 
- Создание программ для iSeries Access для Windows

Отслеживать все изменения в конфигурации PC в процессе установки можно с помощью целого ряда специализированных инструментов. На момент публикации этой информации несколько таких инструментов были доступны на Web-сайтах ZDNet и InstallSite, страница General tools > Analyzing a setup. Эти инструменты и Web-сайты никак не связаны с фирмой IBM.

Примечание: ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

Новое в версии V5R2

Новые функции, доступные администраторам iSeries Access для Windows:

- **Поддержка 64-разрядного ODBC/OLE DB**
В новой версии продукт iSeries Access для Windows поддерживает как 32-, так и 64-разрядный драйвер ODBC. В 64-разрядных версиях Windows 64-разрядный драйвер ODBC автоматически устанавливается вместе с 32-разрядным драйвером. Приложения ODBC, работающие в 64-разрядных версиях Windows, автоматически выбирают нужный драйвер ODBC для каждого приложения. Например, 64-разрядный драйвер применяется только 64-разрядными приложениями. Дополнительная информация приведена в разделе Поддержка 64-разрядного ODBC руководства пользователя iSeries Access для Windows.
- **Индикатор "тихой" установки**
В версии V5R2 у "тихой" установки появился индикатор состояния. Индикатор состояния "тихой" установки представляет собой значок на панели задач, который показан в ходе установки. Индикатор "тихой" установки можно развернуть для просмотра дополнительной информации об установке. Более подробные сведения приведены в разделе "Тихая" установка iSeries Access для Windows.
- **Поддержка Kerberos**
Продукт iSeries Access для Windows теперь позволяет применять для идентификации субъекты и паспорта Kerberos вместо ИД и паролей пользователей. Эта функция доступна в операционных системах Microsoft Windows 2000, XP и более поздних версий, поддерживающих протокол Kerberos. Серверы iSeries версии V5R2 и выше могут применяться в сетях Kerberos. Для этого нужно настроить Службу сетевой идентификации. Дополнительная информация приведена в разделе Служба сетевой идентификации.
- **Усовершенствованная команда PC CWBCFG**
Команда PC CWBCFG теперь позволяет указать каталог, в котором эмулятор PC5250 будет хранить файлы пользователей PC. Дополнительная информация приведена в разделе Задание расположения файлов PC5250 для всех пользователей.
- **Поддержка независимых ASP**
Продукт iSeries Access для Windows теперь поддерживает работу с несколькими базами данных в независимых ASP. Дополнительная информация приведена в разделе Настройка системы для применения драйвера ODBC iSeries Access.
- **При выборочной установке можно установить SSL**
Если специальный установочный образ создается с на базе установочного образа, содержащего

поддержку SSL, то в специальный установочный образ можно включить SSL. Дополнительная информация приведена в разделе Создание специального установочного образа iSeries Access для Windows.

- **Драйвер ODBC iSeries для Linux**

В логическом разделе iSeries можно установить операционную систему Linux и с помощью драйвера ODBC iSeries для Linux обращаться к базе данных iSeries.

Примечание: драйвер ODBC iSeries для Linux не входит в состав продукта iSeries Access для Windows. Это отдельный продукт, применяемый только в операционной системе Linux.

- **Изменения в названиях**

- Служба Удаленная команда Client Access Express теперь называется службой Удаленная команда iSeries Access для Windows.
- Драйвер ODBC Client Access (32-разрядный) теперь носит название Драйвер ODBC iSeries Access. Примечание: для обеспечения совместимости Драйвер ODBC Client Access (32-разрядный) сохранен в текущей версии, однако в конфигурации источников данных следует указать новое имя. Оба названия ссылаются на один драйвер.

- **Больше не поддерживается**

- Не поддерживается 56-разрядное шифрование SSL (CE2). Поддерживается только 128-разрядное шифрование SSL.
- Продукт iSeries Access для Windows версии V5R2 не поддерживает операционную систему Windows 95.

Как напечатать этот раздел

Для просмотра или загрузки версии файла в формате PDF выберите Администрирование iSeries Access для Windows (около 350 Кб или 114 страниц).

Сохранение файлов PDF

Для сохранения PDF-версии на рабочей станции (для дальнейшего просмотра или печати) выполните следующие действия:

1. Щелкните правой кнопкой на файле PDF в окне браузера (щелкните правой кнопкой мыши на приведенной выше ссылке).
2. Выберите **Сохранить как...**
3. Выберите каталог, в котором вы хотите сохранить PDF-версию документа.
4. Нажмите **Сохранить**.

Загрузка программы Adobe Acrobat Reader

Копию программы Adobe Acrobat Reader, необходимую для просмотра и печати документов в формате PDF, можно загрузить с Web-сайта фирмы Adobe

(www.adobe.com/products/acrobat/readstep.html)  .

Сетевые среды iSeries Access для Windows

В iSeries Access для Windows предусмотрено несколько способов подключения клиентов к службам iSeries. Обычно устанавливается прямое соединение между PC, на котором запущен iSeries Access для Windows, и сервером iSeries. Тем не менее, следующие способы позволяют воспользоваться преимуществами других сетевых сред.

- **Microsoft Windows NT 4.0 Terminal Server Edition (TSE), либо Windows 2000 со службами терминала**

TSE - это многопользовательская версия Windows NT server 4.0, которая позволяет запускать на одном сервере Windows NT 4.0 несколько параллельных клиентских сеансов. TSE позволят подключаться клиентам с разных платформ, таких как сетевые станции, UNIX, DOS, OS/2 и многих других. Установка iSeries Access для Windows на сервере TSE позволит предоставить доступ к службам iSeries рабочим станциям, на которых не установлен iSeries Access для Windows. Эти функции также доступны с помощью служб Terminal Services, предусмотренных во всех серверных версиях Windows 2000.

- **iSeries Access для Windows в трехуровневой среде**

Установив iSeries Access для Windows на среднем уровне трехуровневой среды, можно предоставить доступ к службам iSeries широкому диапазону рабочих станций. Кроме того, вы сможете воспользоваться и другими преимуществами трехуровневой среды, в частности, расширенным управлением транзакциями.

С помощью iSeries Access для Windows можно управлять работой многопользовательских PC:

- **Добавление конфигурации TCP/IP всем пользователям**

Для настройки соединений с сервером iSeries для всех пользователей операционной системы Windows NT/2000/XP Workstation или Server введите команду CWBCFG.

- **Пользовательские профайлы для многопользовательских PC**

Операционные системы Windows позволяют применять универсальные мобильные пользовательские профайлы для управления многопользовательскими PC.

Microsoft Windows Terminal Server

Microsoft Windows Terminal Server Edition (TSE) - это многопользовательская версия Windows NT Server 4.0, которая позволяет запускать на одном сервере Windows NT 4.0 несколько параллельных клиентских сеансов. TSE позволяет подключаться клиентам с разных платформ, включая сетевые станции, UNIX, DOS, OS/2 и многие другие. Установка iSeries Access для Windows на сервере TSE позволит предоставить доступ к службам iSeries рабочим станциям, на которых не установлен iSeries Access для Windows. Эти функции также доступны с помощью служб Terminal Services, которые входят во все серверные версии Windows 2000.

Прим.: При работе с Windows 2000, на котором запущены службы Terminal Services, присвойте параметру **Когда проверять уровень обслуживания** на вкладке **Службы** окна свойств iSeries Access для Windows значение **Никогда**.

Дополнительная информация об установке, поддержке, известных неполадках и способах их устранения при работе с iSeries Access для Windows и Microsoft Windows Terminal Server приведена в документе APAR II11373 

Дополнительная общая информация о TSE приведена на Web-сайте Microsoft Windows NT Server 4.0 Terminal Server Edition. 

Применение iSeries Access для Windows в трехуровневой среде

Установка iSeries Access для Windows на среднем уровне трехуровневой среды позволяет предоставить доступ к службам iSeries широкому диапазону рабочих станций. Работа в трехуровневой среде дает и другие преимущества:

- **Усовершенствованная интеграция между отдельными клиентами и приложениями сервера:**

Несколько пользовательских приложений, запущенных на разных клиентах, могут одновременно взаимодействовать с несколькими приложениями на сервере Windows NT/2000. В свою очередь, каждое из нескольких приложений на сервере Windows NT/2000 может взаимодействовать с несколькими базами данных.

- **Расширенное управление транзакциями с помощью сервера Microsoft Transaction Server (MTS):** Трехуровневая среда позволяет выполнять более сложные транзакции, например зависящие друг от друга. (Успешное выполнение транзакции возможно только в случае успешного выполнения всех транзакций.)
- **Импорт данных с сервера iSeries на Web-страницы с применением продукта Microsoft Internet Information Server (IIS):** С помощью Active Server Pages сервер IIS может динамически обновлять Web-страницы на основе информации из DB2 Universal Database для iSeries.

Во всех трехуровневых средах компоненты и приложения разделены на три уровня. Эти уровни могут располагаться на отдельных PC, или терминалах, и взаимодействовать через сеть. Как правило, три уровня характеризуются следующими признаками:

Уровень клиента

На этом уровне работают интерфейсы и приложения, с помощью которых конечные пользователи управляют данными. Примерами могут служить Web-браузер, запущенный на сетевой станции, или пользовательское приложение, использующее удаленные компоненты. Клиенты iSeries Access для Windows на этом уровне не используются.

Средний уровень

На этом уровне находится логика деловых приложений и прикладных программ. В средах, применяющих iSeries Access для Windows, этот уровень должен состоять из сервера Windows, на котором должен быть запущен сценарий Microsoft Active Server Pages или удаленный компонент. Кроме того, на этом уровне используются серверы Microsoft's Internet Information Server (IIS) и Microsoft Transaction Server (MTS) для управления транзакциями между данным уровнем и уровнем клиента. Продукт iSeries Access для Windows поддерживает MTS на клиентских машинах с помощью драйвера ODBC и взаимодействует с уровнем базы данных. В настоящее время для доступа к данным из компонентов, находящихся на среднем уровне, фирма Microsoft рекомендует применять OLE DB, ActiveX Data Objects (ADO) и Remote Data Service.

Дополнительная информация о среднем уровне приведена в следующих разделах:

- MTS
- Доступ к службам iSeries из среднего уровня

Уровень базы данных

Этот уровень обычно состоит из базы данных DB2 Universal Database для iSeries. Приложения могут получить доступ к этой и другим службам iSeries с помощью программ серверов хоста или пользовательских программ iSeries.

Применение Microsoft Transaction Server (MTS)

Клиент iSeries Access для Windows поддерживает MTS версий 2.x и более поздних версий с драйвером ODBC iSeries Access на серверах V5R1 и более поздних выпусков.

MTS

MTS - это многокомпонентная программная модель и среда выполнения фирмы Microsoft, предназначенная для разработки, размещения и управления приложениями сервера для Internet. Во многих трехуровневых средах служба Active Server Pages (ASP) вызывают компоненты MTS для получения доступа к базам данных, приложениям больших систем и очередям заданий. Компоненты MTS применяются в среднем уровне трехуровневой среды iSeries Access для Windows и управляют транзакциями между приложениями клиентов, компонентами iSeries Access для Windows и базами данных, участвующими в транзакциях.

Для управления транзакциями, распределенными между несколькими системами управления базами данных (СУБД), и для обеспечения целостности двухфазовой фиксации при работе с транзакциями, выполнение которых зависит от состояния обеих сторон, MTS применяет Microsoft Distributed Transaction Coordinator (MSDTC).

Замечания по применению

- Если MSDTC не может загрузить драйвер iSeries Access ODBC, то команда SQLSetConnectAttr(SQL_ATTR_ENLIST_IN_DTC) не будет выполнена с кодом причины 2 (сбой XaRmCreate). Если установлена компонент PC5250, то полный путь среды MSDTC будет установлен автоматически. Во избежание такого эффекта в соответствующую переменную среды на PC, на котором запущен MSDTC, необходимо внести путь к каталогу Shared в установочном каталоге iSeries Access для Windows. Например: C:\Program Files\IBM\Client Access\Shared.
- Если в меню **Соединения** → **Свойства** Навигатора указано значение SSL или любое другое настраиваемое значение, то имя соединения iSeries в Навигаторе должно совпадать с именем соединения на клиентском PC, работающем под управлением MTS. Имена соединений MSDTC с базой данных UDB DB2 для iSeries совпадают с именами соединений клиентов ODBC iSeries Access для Windows на PC, работающих под управлением MTS. Для изменения свойств соединений MSDTC необходимо изменить реестр системных учетных записей.

Это можно сделать с помощью утилиты удаленных команд (IRC) в сочетании с утилитой CWBENV:

1. Запустите CWBENV на клиентском PC для получения информации о конфигурации среды.
2. Скопируйте полученные данные на PC MSDTC.
3. Запустите службу обработки удаленных команд iSeries Access for Windows и убедитесь, что она настроена для работы в режиме локальной системы.
4. С помощью команды RUNRMTCMD передайте из сеанса PC5250 на PC команду CWBENV, чтобы импортировать среду.

Дополнительная информация об этих функциях приведена в руководстве пользователя iSeries Access для Windows.

Дополнительная информация о MTS приведена на Web-сайте MTS фирмы Microsoft. 

Обеспечение доступа к службам iSeries компонентам среднего уровня

Существует несколько способов обеспечить доступ к серверу iSeries компонентам среднего уровня.

Примечание: у компонентов среднего уровня нет пользовательского интерфейса, поэтому приложение с трехуровневой архитектурой может зависнуть, если iSeries Access показывает меню входа в систему. Для того чтобы этого не происходило, разработчики приложения должны предоставить всю необходимую информацию о соединении (ИД пользователя и пароль), передав ее серверу iSeries через специальный системный объект. Параметру вывода приглашения, связанному с этим объектом, нужно присвоить значение **никогда не выводить приглашение**.

Комплекс связи OLE DB iSeries Access для Windows

Для работы с комплексом связи OLE DB iSeries Access для Windows большинство приложений применяют объекты данных ActiveX (ADO). Ниже описаны преимущества такого подхода:

- Для работы с программами, командами, запросами SQL, хранимыми процедурами, физическими и логическими файлами системы iSeries разработчикам придется лишь в минимальной степени изменить существующие интерфейсы и технологию создания программ.
- Такой подход обеспечивает автоматическое преобразование типов данных, применяемых в системе iSeries и PC.

- Он позволяет снизить нагрузку на систему при работе с SQL, поддерживая доступ к файлам на уровне записей.
- С его помощью достаточно просто разрабатывать и создавать приложения. Обычно такой подход обеспечивает наиболее простой способ создания приложений с трехуровневой архитектурой.

Дополнительная информация по этому вопросу приведена в разделе Создание программ OLE DB.

Драйвер ODBC iSeries Access для Windows

С помощью комплекса связи Microsoft OLE DB для ODBC (MSDASQL) можно обращаться к драйверу ODBC iSeries Access через объекты ADO и RDS. Применение драйвера ODBC iSeries Access с комплексом связи OLE DB iSeries Access для Windows предоставляет вам следующие возможности:

- Более широкий набор возможностей SQL
Драйвер ODBC позволяет применять обновляемые курсоры, управление фиксацией SQL и хранимые процедуры с несколькими наборами результатов.
- Пул соединений
В большинстве приложений MTS и ASP для обработки каждого запроса клиента устанавливается отдельное соединение с сервером iSeries. Драйвер ODBC позволяет создать пул постоянных соединений. Поскольку часто большая часть ресурсов тратится на запуск задания iSeries, а не на выполнение запроса, такой пул позволяет значительно повысить скорость обработки запросов.

Дополнительная информация о работе с ODBC с помощью ADO приведена в разделе Выбор интерфейса для работы с драйвером ODBC.

Прочая информация о драйвере ODBC iSeries Access приведена в разделе Программирование ODBC.

Примечание: Для работы комплекса связи OLE DB iSeries Access для Windows и некоторых функций драйвера ODBC iSeries Access необходима программа MDAC 2.5 или более поздней версии. За дополнительной информацией обратитесь к описанию требований MDAC.

Автообъект ActiveX

Клиент iSeries Access для Windows предоставляет разработчикам библиотеку новых автообъектов ActiveX для компонентов среднего уровня. Такие объекты позволяют получить доступ к:

- Очередям данных iSeries
- Удаленным командам и вызовам распределенных программ
- Объектам администрирования
- Системным объектам iSeries
- Доступ к таблицам базы данных iSeries для функции Передача данных

В некоторых случаях объекты ActiveX предоставляют больший набор возможностей, чем объекты ADO, однако для работы с ними требуется применять более сложные средства программирования.

Примечание: клиент iSeries Access для Windows содержит библиотеку автообъектов, разработанную для клиента Windows 95/NT (продукт XD1). Эти автообъекты, в том числе база данных, не адаптированы для трехуровневой архитектуры.

API C/C++ в Client Access Express

API iSeries Access для Windows обеспечивают эффективный низкоуровневый доступ к серверам хоста OS/400. Однако для применения таких API требуется определенный опыт создания программ на языках C/C++. В частности, вы должны быть знакомы с API и типами данных языка C, а также с правилами создания компонентов с поддержкой нескольких нитей.

Добавление всех пользователей в конфигурацию TCP/IP

Для настройки соединений с сервером iSeries для всех пользователей операционной системы Windows NT/2000/XP Workstation или Server введите в командной строке системы Windows NT/2000/XP команду CWBCFG. Помимо существующих пользователей, соединения будут настроены и для пользователя Windows по умолчанию - профайла, который применяется по умолчанию при создании новых пользователей в системе Windows NT/2000/XP.

Дополнительная информация о команде CWBCFG приведена в электронном руководстве пользователя iSeries Access для Windows.

Задание расположения файлов PC5250 для всех пользователей

Для того чтобы задать каталог, в котором эмулятор PC5250 будет искать и сохранять файлы для всех пользователей, введите команду CWBCFG в командной строке Windows NT/2000/XP. Если этот каталог не задан с помощью команды CWBCFG, то по умолчанию применяется подкаталог \emulator\private установочной папки iSeries Access для Windows. Этот каталог применяется всеми пользователями PC совместно, однако не у всех пользователей могут быть права на запись в этот каталог.

Поскольку команда CWBCFG задает параметры для пользователя Windows по умолчанию, то все учетные записи пользователей, созданные после выполнения команды CWBCFG, будут применять каталог, заданный с помощью команды CWBCFG, вместо указанного выше стандартного каталога.

Дополнительная информация о команде CWBCFG приведена в электронном руководстве пользователя iSeries Access для Windows.

Пользовательские профайлы для многопользовательских PC

Продукт iSeries Access для Windows может применяться на PC, с которыми работают несколько пользователей. Такая функция администрирования поддерживается в операционных системах Windows за счет переносимых, универсальных и неизменяемых профайлов.

Прим.: Инструкции по применению этих способов администрирования многопользовательских систем приведена в документации к пакету Microsoft Resource Kit для применяемой операционной системы Windows. Комплекты ресурсов создаются фирмой Microsoft и входят в комплект поставки Microsoft Developers Kit.

Переносимые пользовательские профайлы

Переносимыми называются пользовательские профайлы Windows 95/98/Me, которые можно перемещать с одного PC с Windows 95/98/Me на другой. Такая информация, как параметры рабочего стола, пункты меню Пуск и содержимое системного реестра, хранится в домашнем каталоге пользователя на файловом сервере. Переносимые пользовательские профайлы можно перемещать только между PC, на которых установлена операционная система Windows 95/98/Me.

Универсальные пользовательские профайлы

Универсальными называются пользовательские профайлы Windows NT/2000/XP, которые можно перемещать с одного PC на другой. После перемещения сохраняются все параметры конфигурации. Универсальные профайлы обычно хранятся на сервере NT/2000/XP. В параметрах универсального пользовательского профайла указан путь к каталогу данного пользователя на сервере NT/2000/XP (он называется путем к пользовательскому профайлу). Здесь хранятся данные системного реестра, параметры рабочего стола и пункты меню Пуск этого пользователя. Универсальные пользовательские профайлы можно перемещать только между PC, на которых установлена операционная система Windows NT/2000/XP.

Неизменяемые пользовательские профайлы

Неизменяемыми называются создаваемые системным администратором профайлы пользователей, параметры которых не могут или не должны изменяться. Неизменяемые пользовательские профайлы также можно перемещать с одного PC на другой.

Установка и обновление программы на нескольких PC

Существует несколько способов установить программу iSeries Access для Windows на нескольких PC, не повторяя на каждом компьютере всю процедуру установки и настройки. При этом вы можете ограничить набор функций, доступных пользователям, указав, какие компоненты должны быть установлены на этих компьютерах.

Информация

• Требования к MDAC

В отличие от предыдущих версий, теперь вместе с iSeries Access для Windows не устанавливается MDAC. Однако следует учитывать, что драйвер ODBC iSeries Access и комплекс связи OLE DB iSeries Access для Windows предъявляют ряд требований к уровню MDAC на PC. Перед установкой продукта iSeries Access для Windows в среде Windows 98/NT/Me нужно убедиться в наличии требуемого уровня MDAC. В операционных системах Windows 2000 и более поздних версиях необходимый уровень компонентов MDAC уже установлен.

Требуемые уровни MDAC:

- Драйвер ODBC iSeries Access - MDAC 2.5 или более поздней версии для поддержки пула соединений и MTS
- Комплекс связи OLE DB iSeries Access - MDAC 2.5 для всех функций

При отсутствии в системе компонентов MDAC 2.5 или выше iSeries Access для Windows не позволит установить компонент OLE DB. Если ранее была выполнена обычная установка старой версии iSeries Access для Windows, а сейчас требуется перейти к версии V5R2M0, то компонент OLE DB будет удален из системы в случае, если перед обновлением не будет установлена программа MDAC 2.5. Продукт MDAC последней версии можно загрузить со следующего

Web-сайта фирмы Microsoft: <http://www.microsoft.com/data>  .

• Перенос данных

Продукт iSeries Access для Windows поддерживает перенос данных только из следующих продуктов:

- Client Access Enhanced для Windows 3.1 (XK1)
- Client Access для Windows 95/NT (XD1), V3R2M0

Ниже описаны некоторые варианты установки:

• Создание специального установочного образа

Можно создать специальный установочный образ, исключив ненужные компоненты из основного установочного образа. Полученный установочный образ может применяться для установки программы на всех компьютерах, подключенных к сети.

• "Тихая" установка и обновление

Вы можете создать файл ответов, содержащий список опций, которые должны быть выбраны в ответ на приглашения программы установки. С помощью полученного файла ответов можно выполнить идентичную установку программы на всех компьютерах, не требующую вмешательства пользователя.

В ходе установки используются файлы из разных каталогов. Поиск необходимых файлов выполняется в подкаталогах каталога ProdData. Дополнительная информация приведена в разделе поиск файлов.

Существует несколько программ, отслеживающих изменения, внесенные программой установки.

Некоторые из таких программ можно загрузить с Web-сайтов ZDNet  и InstallSite , ссылки на которые приведены на странице **Общие инструменты** → **Анализ установки**. Фирма IBM не несет ответственности за работу этих программ и Web-сайтов.

Создание специального установочного образа iSeries Access для Windows

В некоторых случаях требуется ограничить набор компонентов iSeries Access для Windows, доступных для установки. Для этого можно исключить некоторые компоненты из установочного образа, а затем разослать его пользователям. Мастер Специальный установочный образ поможет вам выполнить эту задачу.

Запуск мастера Специальный установочный образ

Для запуска мастера создания специального установочного образа воспользуйтесь компакт-диском iSeries Setup and Operations, либо перейдите в каталог \QIBM\ProdData\Access\Windows\Install\Image (в нем содержится установочный образ) и введите команду cwbinimg.

Работа с установочными образами

Специальные установочные образы не обновляются, когда на сервере iSeries применяются или удаляются временные исправления программ (PTF). Для применения пакетов обслуживания вам потребуется повторно создать установочный образ. Вместо этого пакет обслуживания можно совместить с существующим специальным установочным образом. Соответствующие инструкции приведены на Web-сайте iSeries Access <http://www-1.ibm.com/servers/eserver/iseries/access/> . Выберите самый новый пакет обслуживания и откройте папку subcomps/. Прочтите инструкции в файле Readme.1st, который находится в этой папке.

Рассылка установочного образа

Мастер позволяет указать каталог, в котором должен быть создан специальный установочный образ. Этот каталог должен быть пустым (нельзя заменить старый установочный образ на новый) и не должен совпадать с корневым каталогом. Обратите внимание на то, что программу создания специальных установочных образов содержит только полный установочный образ. Следовательно, этот мастер не будет скопирован на пользовательские PC. Специальный образ можно также скопировать на компакт-диск. Программа установки iSeries Access для Windows будет автоматически запускаться при помещении этого компакт-диска в дисковод CD-ROM.

Примечание: Если на сервере iSeries для iSeries Access для Windows установлено несколько дополнительных языков, то в качестве основного языка нового установочного образа можно выбрать любой из этих языков или основной язык сервера iSeries. Однако в том случае, когда мастер был запущен с компакт-диска, язык выбрать нельзя, так как компакт-диск не содержит поддержку дополнительных языков.

Добавление Secure Sockets Layer (SSL) в установочный образ

Если специальный установочный образ создается с помощью установочного образа, содержащего поддержку SSL, то в специальный установочный образ можно включить SSL. При обнаружении продукта SSL программа специальной установки укажет его в списке компонентов. Выберите этот компонент, если нужно включить поддержку SSL в специальный установочный образ.

Примечание: на SSL распространяются экспортные ограничения США. Вы несете ответственность за соответствие создаваемого установочного образа экспортному законодательству США.

"Тихая" установка iSeries Access для Windows

"Тихой" называется установка iSeries Access для Windows, в процессе которой не требуется никакого вмешательства пользователя. Все параметры установки указаны в файле ответов, поэтому при установке не появляются никакие окна и запросы. Для выполнения "тихой" установки:

1. Создайте файл ответов.
2. Запустите "тихую" установку.
3. Проверьте, все ли компоненты установлены, просмотрев коды возврата в файле протокола.

Примечание: хотя процедура "тихой" установки с переходом к новой версии схожа с процедурой "тихой" установки, в них применяются разные процедуры создания файла ответов.

Файл ответов содержит опции установки, которые в случае обычной установки вам пришлось бы вводить вручную в ответ на запросы системы.

Индикатор "тихой" установки

В процедуре "тихой" применяется индикатор состояния. Индикатор состояния "тихой" установки представляет собой значок на панели задач, который показан во время "тихой" установки. Если подвести курсор мыши к этому значку, то будет показан объем выполненной работы в процентах от общего объема. Кроме того, индикатор "тихой" установки можно развернуть для просмотра дополнительной информации. После успешного завершения установки этот значок исчезнет с панели задач. Если при установке возникнут ошибки, значок останется на панели задач и будет помечен небольшим красным треугольником. Щелкнув на красном треугольнике, можно просмотреть сведения об ошибках.

Примечания:

- Если значение индикатора "тихой" установки не изменяется слишком долго, проверьте файл протокола на наличие сообщений об ошибках.
- Для устранения сбоя "тихой" установки рекомендуется запустить программу установки на PC в обычном режиме и проверить, не появляются ли какие-либо нестандартные окна до окна Подтверждение выбранных компонентов. Большинство сбоев "тихой" установки, как правило, бывает вызвано появлением нестандартных окон до начала копирования установочных файлов компонентов.

Различия между обычной и "тихой" установками

В следующей таблице описаны различия между обычной и "тихой" установками, заключающиеся в различных способах обработки одних и тех же ситуаций, возникающих в процессе установки.

Событие	При обычной установке...	При "тихой" и регистрируемой установке...
Выберите компонент Эмулятор дисплея и принтера 5250 или Консоль управления AS/400 для установки в Windows 95/98.	Вы можете указать, следует ли добавить в файл autoexec.bat путь к программе-эмулятору.	Путь к программе-эмулятору будет добавлен в файл autoexec.bat автоматически.

Событие	При обычной установке...	При "тихой" и регистрируемой установке...
На PC установлен продукт Client Access для Windows 95/NT Lightning SDK и во время установки iSeries Access для Windows был выбран компонент Мастеры Visual Basic.	Во время установки появится предупреждение о том, что установка компонента Мастеры Visual Basic приведет к удалению продукта Client Access для Windows 95/NT Lightning SDK.	Предупреждение не появляется, и Client Access для Windows 95/NT Lightning SDK автоматически удаляется при установке Мастеров Visual Basic.
Попытка установки компонента, который запрещен (согласно стратегии, из-за зависимости от другого компонента или в силу иных причин) или несовместим с установленным в системе компонентом.	Появится окно диалога со списком всех компонентов, запрещенных по указанным причинам. Компонент не будет установлен.	Компонент не будет установлен.
Выполните "тихую" установку с переходом к новой версии Client Access для Windows 95/NT.	По умолчанию установка будет выполнена в установочном каталоге Client Access для Windows 95/NT. Вы можете указать и другой каталог для установки, но при этом может быть потеряна некоторая информация о конфигурации исходной системы. Появится соответствующее предупреждающее сообщение.	По умолчанию для установки iSeries Access для Windows будет выбран установочный каталог Client Access для Windows 95/NT (вне зависимости от каталога, заданного в файле ответов).
Произошла ошибка.	Выдаются сообщения об ошибках.	Сообщения об ошибках выдаются во время регистрируемой установки и не выдаются во время "тихой" установки. В файл протокола "тихой" установки заносится отрицательное число, что означает, что во время установки произошли ошибки. Если во время "тихой" установки возникают неполадки, выполните установку в интерактивном режиме и убедитесь, что неполадка не связана с выбором "тихого" режима.

Создание файлов ответов для установки iSeries Access для Windows

В файлах ответов хранится список опций, которые должны быть выбраны в ответ на приглашения, выводимые во время установки. Такие файлы применяются во время "тихой" установки для получения информации, необходимой для выполнения установки.

Для создания файла ответов выполните следующие действия:

1. Перейдите в каталог PC, содержащий установочный образ iSeries Access для Windows, и введите следующую команду в командной строке:

```
setup -r -f1d:\dir\file.iss
```

Эта команда запускает установку и позволяет записать ответы.

- **-f1** - это необязательный параметр, позволяющий задать альтернативный файл ответов. Если вы не укажете этот параметр, то все ответы будут записаны в файле setup.iss. Файл Setup.iss находится в каталоге Windows, например, C:\Windows или C:\Winnt, в зависимости от установленной операционной системы.
- **d:\dir** - путь к каталогу, в котором должен быть создан файл ответов. Если задан параметр **-f1**, то вместе с каталогом должен быть задано имя файла ответов.

- **file.iss** - имя файла ответов, который должен быть создан. Файлы ответов всегда имеют расширение iss.
2. Ответьте на вопросы программы установки - указанные ответы будут использованы во время "тихой" установки.

После завершения установки будет создан файл .iss, аналогичный тому, который приведен в качестве примера файла ответов.

Запуск "тихой" установки

В процессе "тихой" установки для ответа на вопросы программы установки применяется файл ответов (file.iss). Это позволяет выполнять установку без вмешательства пользователя и быстро передавать установочные образы по сети. Информацию о состоянии "тихой" установки можно заносить в файл протокола (file.log).

Для запуска "тихой" установки введите в командной строке установочного каталога iSeries Access для Windows следующую команду:

```
setup -s -f1d:\dir\file.iss -f2d:\dir\file.log
```

где:

- **-f1** - необязательный параметр, в котором указывается имя файла ответов (**file.iss**). Если этот параметр не указан, программа установки попытается использовать файл ответов по умолчанию - setup.iss. Она будет искать этот файл в каталоге, в котором находится программа setup.exe. (Вместо **d:\dir** подставьте диск и каталог, в котором находится ваш файл ответов.) Вместе с опцией **-f1** необходимо задать полное имя файла ответов.
- **-f2** - это необязательный параметр, в котором указывается полное имя файла протокола, который будет создан во время "тихой" установки. Если он не указан, программа установки создаст файл протокола с именем setup.log и поместит его в каталог, в котором находится файл setup.exe. (Здесь **d:\dir** обозначает диск и каталог, содержащий файл протокола.) Вместе с опцией **-f2** необходимо задать полное имя файла протокола. Вместо **file.log** укажите имя создаваемого файла протокола.

Коды возврата для "тихой" установки или перехода

Для того чтобы узнать, успешно ли была выполнена "тихая" установка, просмотрите коды возврата в файле протокола. Код 0 означает, что установка была выполнена успешно. Ненулевой код означает, что при установке возникли неполадки, которые необходимо устранить. Имя и расположение файла протокола вы указали при запуске "тихой" установки. Кроме того, дополнительная информация о сбоях сохраняется в файле silent.txt в целевом каталоге, а до создания целевого каталога - в файле cwbsilent.txt в каталоге Windows (Windows или Winnt).

код возврата	значение
0	Успешно
-1	Общая ошибка
-2	Недопустимый режим
-3	В файле Setup.iss отсутствуют необходимые данные
-4	Недостаточно памяти
-5	Файл не существует
-6	Файл ответов недоступен для записи
-7	Файл протокола недоступен для записи
-8	Неверно указан путь к файлу ответов InstallShield для "тихой" установки
-9	Неверный тип списка (строка или число)
-10	Неверный тип данных

код возврата	значение
-11	Неизвестная ошибка при установке
-12	Нарушена последовательность окон диалога
-51	Не удается создать указанную папку
-52	Указанный файл или папка недоступны
-53	Выбрана неверная опция

Дополнительная информация приведена в разделе Запуск "тихой" установки.

Работа с пакетами обслуживания

Пакеты обслуживания содержат исправления программы iSeries Access для Windows. Они поставляются в виде временных исправлений программы (PTF). Для повышения надежности работы клиента iSeries Access для Windows и исправления известных неполадок рекомендуется регулярно загружать новые PTF на сервер iSeries. После установки PTF на хосте вызовите функцию Проверить уровень обслуживания для рассылки пакетов обслуживания на PC клиентов.

Загрузка нового PTF и его установка на сервере iSeries

Для заказа PTF на сервере iSeries служит команда SNDPTFORD. Поскольку обычно размер PTF с пакетами обслуживания превышает максимальный размер данных, которые могут быть переданы по сети, рекомендуется разрешить отправку PTF на носителе данных. Для этого укажите в параметре DELIVERY (Способ доставки) команды SNDPTFORD значение *ANY. (По умолчанию параметру присваивается значение *LINKONLY.) Кроме того, вы можете воспользоваться службой доставки PTF по Internet (iPTF). Для просмотра дополнительной информации об этой службе и описания ресурсов,

необходимых для работы с ней, откройте Web-страницу технической поддержки iSeries  и выберите в меню пункт **Fixes and Updates**.

Установка пакетов обслуживания непосредственно на клиентские PC

Пакеты обслуживания можно загрузить на клиентские PC напрямую. Таким образом вы можете обновить программу на клиентских PC, не применяя PTF на сервере. Для получения новой версии

пакета обслуживания откройте домашнюю страницу iSeries Access  и перейдите по ссылке **Service Pack**. После загрузки пакета обслуживания запустите файл setup - обновление будет выполнено автоматически. После установки пакета обслуживания вам потребуется перезагрузить систему.

При установке PTF с пакетом обслуживания обновляется установочный образ iSeries Access для Windows, хранящийся на сервере iSeries. Установка любого PTF повлияет на уровень пакета обслуживания, установленного на сервере iSeries.

Примечание: в системе Windows NT/2000/XP для установки пакета обслуживания и обновления программы iSeries Access для Windows необходимы права системного администратора. Вы можете отменить такое ограничение, применяемое в Windows NT/2000/XP по умолчанию, разрешив устанавливать пакеты обслуживания пользователям без полномочий системного администратора.

Обслуживание иных компонентов и приложений других фирм

Функция проверки уровня обслуживания поддерживает и другие компоненты, например, SSL, а также приложения других фирм (встраиваемые и дополнительные модули). Эта функция автоматически проверяет наличие обновлений для установленных компонентов на сервере iSeries. При обнаружении таких обновлений пользователю обычно отправляется извещение с предложением

установить обновление. Если пользователь разрешает установку, запускается программа Выборочная установка в особом режиме, которая обновляет соответствующий компонент.

Проверка уровня обслуживания

Функция проверки уровня обслуживания iSeries Access для Windows предназначена для получения обновлений программы iSeries Access для Windows и ее компонентов с сервера iSeries. Для того чтобы задать параметры проверки обслуживания, откройте страницу **Обслуживание** в окне **Свойства iSeries Access для Windows**.

На этой вкладке можно задавать следующие параметры:

- Время проверки уровня обслуживания
- Дату проверки уровня обслуживания
- Периодичность проверки уровня обслуживания
- Интервал между входом в систему и проверкой уровня обслуживания в минутах

Примечание Возможно, некоторые из указанных значений задаются стратегиями. Например, в стратегии может быть определен период проверки. В этом случае изменить его нельзя. Настраивать опции для приведенных выше функций можно также с помощью компонента Администрирование приложений.

Кроме того, вы можете выбрать опцию проверки уровня обслуживания в фоновом режиме.

"Тихая" установка пакета обслуживания

Для "тихой" проверки уровня обслуживания и установки пакетов обслуживания необходимо отметить переключатель **Выполнять "тихую" установку** на вкладке **Службы** окна **Свойства iSeries Access для Windows**. Утилита "тихой" установки пакета обслуживания автоматически отвечает на вопросы процедуры установки, основываясь на информации из файла ответов.

Файл ответов схож с тем, который применяется при "тихой" установке, за тем исключением, что необходимо задать следующее имя:

- SLTSP.ISS - для пакетов обслуживания (этот файл должен находиться в том же каталоге, что и файл **setup.exe** данного пакета обслуживания)
- SLTUP.ISS - для модернизаций (этот файл должен находиться в том же каталоге, что и файл **setup.exe** программы установки)

В файле ответов можно указать параметр для автоматической перезагрузки системы. Если вы установите эту опцию, то рекомендуется с помощью параметра SCHEDCHECK запланировать "тихую" проверку версии обслуживания на ночное время. Дополнительную информацию о параметре SCHEDCHECK можно найти в электронном руководстве пользователя iSeries Access для Windows.

Если вы не выберете эту опцию, то вам будет предложено

Администрирование ODBC

Открытая связь с базами данных (ODBC) - это стандарт фирмы Microsoft для обеспечения доступа к базам данных. Он содержит набор явно заданных интерфейсов прикладных программ (API), применяющих язык структурных запросов (SQL) для доступа к базам данных.

Обзор драйвера ODBC iSeries Access

Этот раздел содержит общую информацию о драйвере ODBC и его применении в iSeries Access для Windows.

Настройка системы для применения драйвера ODBC

В этом разделе приведено описание действий по настройке среды системы для поддержки

драйвера ODBC. Для просмотра справки по настройке драйвера ODBC запустите программу администрирования ODBC в группе программ iSeries Access для Windows и обратитесь к электронной справке.

Вопросы защиты ODBC

В этом разделе рассмотрено несколько вопросов защиты при работе с ODBC и приведены ссылки на более подробные руководства по защите.

Устранение неполадок ODBC

В этом разделе приведены рекомендации по устранению наиболее часто встречающихся неполадок при работе с iSeries Access для Windows и ODBC. В нем также рассматриваются несколько инструментов, которые могут пригодиться при оптимизации производительности. Перед обращением в службу технической поддержки рекомендуется просмотреть этот раздел.

Драйвер ODBC iSeries для Linux

Этот раздел содержит информацию об установке операционной системы Linux в логическом разделе iSeries и применении драйвера ODBC iSeries для Linux для работы с базой данных iSeries.

Примечание: драйвер ODBC iSeries для Linux не входит в состав продукта iSeries Access для Windows. Это отдельный продукт, применяемый только в операционной системе Linux.

Инструкции по настройке поддержки ODBC в пользовательских приложениях приведены в разделе Программирование ODBC. Этот раздел состоит из нескольких частей:

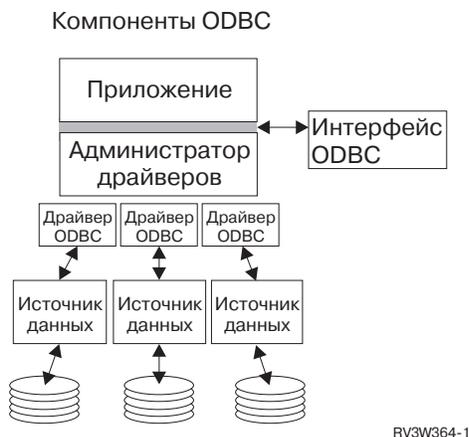
- Список API ODBC
- Реализация API ODBC
- Примеры программ
- Производительность ODBC

Обзор драйвера ODBC iSeries Access

Драйвер ODBC iSeries Access - это совокупность интерфейсов прикладных программ (API), предназначенных для доступа к базам данных с помощью языка структурных запросов (SQL). Применение этого драйвера позволяет приложениям получать доступ к различным базам данных сервера iSeries с помощью одного и того же исходного кода и обрабатывать данные в наиболее удобном для себя формате. ODBC предоставляет разработчику приложений относительно простую модель для создания переносимых приложений или компонентов, которые должны взаимодействовать с несколькими СУБД.

Архитектура ODBC состоит из приложения, диспетчера драйверов, драйвера ODBC и источника данных. Продукт iSeries Access обеспечивает поддержку как 32-разрядного, так и 64-разрядного драйвера ODBC. В 64-разрядных версиях Windows 64-разрядный драйвер ODBC автоматически устанавливается вместе с 32-разрядным драйвером. Приложения ODBC, работающие в 64-разрядных версиях Windows, автоматически выбирают нужный драйвер ODBC для каждого приложения. Например, 64-разрядный драйвер применяется только 64-разрядными приложениями.

Для применения ODBC нужно настроить источник данных. Это можно сделать с помощью Администратора ODBC. В папке iSeries Access для Windows предусмотрено две версии Администратора: 32-разрядная и 64-разрядная. С помощью Администратора ODBC можно настраивать три вида источников данных: пользовательский, системный и файловый. Дополнительная информация о настройке источников данных приведена в разделе Поддержка 64-разрядного ODBC руководства пользователя iSeries Access для Windows.



Приложение выполняет обработку и вызывает функции ODBC для выполнения операторов SQL.

Диспетчер драйвера обрабатывает вызовы функций ODBC и передает запросы драйверу.

Драйвер обрабатывает вызовы функции ODBC, передает запросы SQL на выполнение к источнику данных и возвращает результат приложению.

Источник данных - для применения источника данных необходимо создать имя источника данных (DSN). DSN содержит информацию о возможных способах доступа к СУБД. Можно задать любое из приведенных ниже DSN:

- *Пользовательское DSN:* Эти источники данных являются локальными для каждого компьютера и доступны только своему создателю. Эта информация хранится в реестре.
- *Системное DSN:* Эти источники данных являются локальными для каждого компьютера, не будучи выделенными какому-либо пользователю. Присвоить источнику данных системное DSN может система или пользователь с соответствующими правами доступа. Эта информация хранится в реестре.

Примечание: на PC с 64-разрядными процессорами системная часть реестра разделена на 32-разрядный и 64-разрядный разделы. Системные DSN, настроенные с помощью 32-разрядного Администратора ODBC, поддерживаются только 32-разрядными приложениями. В свою очередь, системные DSN, настроенные с помощью 64-разрядного Администратора ODBC поддерживаются только 64-разрядными приложениями.

- *Файловое DSN:* Эти источники данных основаны на файлах, общих для всех пользователей с одинаковыми драйверами, предоставляющими им доступ к базе данных. Эти источники данных не обязательно являются локальными или выделенными какому-либо пользователю.

Дополнительная информация по ODBC приведена на Web-сайте фирмы Microsoft.

Настройка системы для применения драйвера ODBC iSeries Access

Данный драйвер ODBC iSeries Access совместим с драйвером ODBC версии 3.5. Для работы этого драйвера необходим продукт MDAC версии 1.5 или выше. Для приложений, применяющих объекты данных ActiveX (ADO), необходим продукт MDAC версии 2.1 или выше. Библиотеки MDAC версии 2.1 и более поздних версий обеспечивают дополнительные функции доступа к данным iSeries для приложений, применяющих ADO, комплекс связи ODBC фирмы Microsoft и ODBC iSeries Access для Windows ODBC. В случае, если приложение применяет пул соединений или сервер Microsoft Transaction Server (MTS), рекомендуется установить самую новую версию MDAC. Продукт MDAC

можно загрузить со следующего Web-сайта фирмы Microsoft: <http://www.microsoft.com/data> .

Перед настройкой драйвера ODBC необходимо настроить систему.

Инструкции о настройке системы для применения драйвера ODBC iSeries Access приведены в следующих разделах:

1. Добавьте локальную систему в каталог реляционной базы данных (RDB) в OS/400.
2. Настройте источник данных ODBC.

Следующие инструкции не обязательны для выполнения и могут содержать не все необходимые действия:

Независимые ASP

Для работы с независимыми ASP с помощью ODBC настройте DSN ODBC и выполните следующие действия:

1. Выберите вкладку **Сервер**.
2. Укажите **Имя RDB**, соответствующее независимому ASP, с которым необходимо установить соединение.
3. Если имя RDB не указано, применяется имя RDB по умолчанию, указанное в описании задания в пользовательском профайле, под управлением которого устанавливается соединение ODBC. По умолчанию, драйвер применяет параметр профайла пользователя, устанавливающего соединение ODBC.

Дополнительная информация о независимых ASP приведена в разделе Независимые ASP.

Если вам требуется справочная информация по настройке опций определенного источника данных, откройте раздел Администратор ODBC в группе программ iSeries Access для Windows, выберите нужный источник данных и просмотрите электронную справку.

Добавление локальной системы в каталог RDB

Для применения ODBC в локальной системе она должна быть внесена в каталог RDB.

Для добавления локальной системы в каталог RDB выполните следующие действия:

1. В командной строке введите команду CL Добавить запись каталога реляционной базы данных (ADDRDBDIRE).
2. После того как вам будет предложено ввести значения в меню ADDRDBDIRE, введите в параметре Реляционная база данных имя системы.
3. В параметре Удаленное расположение введите значение *LOCAL.

В случае, если приложение работает с данными в независимых ASP в системе версии V5R2 или выше, для получения набора имен базы данных (RDB) могут понадобиться дополнительные действия. Имя RDB соответствует пространству имен, состоящему из системного ASP и пользовательских ASP, либо подключенной группы ASP, связанной с системным ASP. Дополнительная информация о независимых ASP приведена в разделе Независимые ASP.

Примечание:

ODBC допускает применение полных имен в формате [имя каталога].[имя схемы].идентификатор (идентификатор может быть именем таблицы, представления, процедуры и т.п.). В реализации SQL для DB2/400 это соответствует формату [RDB name].[имя набора].идентификатор.

Выбор источника данных ODBC

Для того чтобы приложение могло обращаться к данным и работать с ними, необходимо настроить источник данных для приложения.

Для указания источника данных выполните следующие действия:

1. Запустите программу администрирования ODBC из группы программ iSeries Access для Windows.
2. Выберите вкладку для нужного типа источника данных. Дополнительная информация приведена в разделе Обзор ODBC.
3. Выберите источник данных из списка или нажмите **Добавить** для создания нового. Для применения существующего источника данных нажмите **Настроить** и перейдите к шагу 5.
4. Выберите драйвер iSeries Access ODBC для источника данных и нажмите кнопку **Готово**.
Примечание: список драйверов содержит 'Драйвер ODBC Client Access (32-разрядный)'. Этот пункт включен для поддержки источников данных, созданных с помощью предыдущих версий Client Access. Оба имени ссылаются на один драйвер ODBC. Вы можете выбрать любое из этих имен, однако в последующих выпусках имя 'Драйвер ODBC Client Access (32-разрядный)' будет удалено.
5. Выберите нужные опции в окне диалога Настройка ODBC для iSeries Access для Windows. Описание управляющих элементов приведено в электронной справке источника данных. Для просмотра справочной информации нажмите клавишу F1 или кнопку Справка.

Примечание:

Имя источника данных может содержать до 32 символов. Оно должно начинаться с алфавитного символа и не может содержать следующие символы:

Недопустимые символы в имени источника данных

Открывающая квадратная скобка ([)

Вопросительный знак

Закрывающая квадратная скобка (])

Звездочка (*)

Открывающая фигурная скобка ({)

Знак равенства (=)

Закрывающая фигурная скобка (})

Восклицательный знак (!)

Открывающая круглая скобка (()

Знак "at" (@)

Открывающая круглая скобка ())

Точка с запятой (;)

Защита ODBC iSeries Access для Windows

Следующая информация не может служить исчерпывающим руководством по стратегиям защиты системы iSeries с продуктом iSeries Access для Windows или без него. Здесь приведен лишь краткий обзор принципов организации защиты при работе с iSeries Access для Windows и ODBC. Более

подробная информация приведена в книге IBM Security - Reference  .

- Рискованные стратегии защиты ODBC
- Стратегии защиты программ ODBC
- Прочая информация по защите ODBC

Рискованные стратегии защиты ODBC

Некоторые системные администраторы уделяют гораздо большее внимание защите способов и средств доступа к данным, нежели защите самих данных. Это весьма рискованно, поскольку в этом случае требуется предусмотреть ВСЕ возможные пути доступа пользователей к данным. Ниже описаны некоторые часто встречающиеся приемы защиты ODBC, которых следует избегать:

Защита командной строки

Такая организация защиты может применяться для стандартных меню iSeries и приложений эмулятора 5250. Предполагается, что если запретить пользователям вводить команды в сеансе эмуляции 5250, то они смогут получать доступ к данным только через программы и меню, которые им предоставит системный администратор. По этой причине такой способ не обеспечивает подлинной защиты. Повысить надежность защиты можно с помощью стратегий iSeries Access и функции администрирование приложений, а также прав доступа на уровне объектов.

Стратегии iSeries Access для Windows позволяют ограничить доступ ODBC к конкретному источнику данных, и при необходимости разрешить только чтение данных. Кроме того, доступ к ODBC можно запретить с помощью функции Администрирование приложений Навигатора iSeries.

Дополнительная информация приведена в книге IBM Security - Reference  .

Пользовательские программы выхода

Пользовательские программы выхода позволяют системному администратору защитить программу сервера хоста, поставляемую фирмой IBM. Драйвер ODBC iSeries Access использует сервер базы данных хоста, в частности точки выхода QIBM_QZDA_INIT, QIBM_QZDA_NDBx и QIBM_QZDA_SQLx. Некоторые драйверы ODBC и способы доступа к данным iSeries Access для Windows (такие как OLE DB) могут использовать и другие серверы хоста.

Журналы

Для обеспечения управления фиксацией в приложениях типа клиент-сервер часто применяется ведение журналов. Журналы содержат подробную информацию о каждом обновлении файла, для которого ведется журнал. Содержимое журнала можно отформатировать. Журналы позволяют получать следующую информацию:

- Профайлы пользователей, обновлявшие файл
- Записи, которые были обновлены
- Тип обновления

Ведение журнала позволяет также создавать пользовательские записи журнала. В сочетании с пользовательскими программами выхода или триггерами это дает сравнительно простой способ пользовательского контроля за системой. Дополнительная информация приведена в книге Резервное

копирование и восстановление  .

Ограничения на Имя источника данных (DSN)

В драйвере ODBC iSeries Access предусмотрен параметр DSN, позволяющий предоставлять доступ к базе данных только для чтения. Драйвер ODBC iSeries Access поддерживает значения "только для чтения" и "для чтения и вызова". Применение этих параметров помогает предотвратить несанкционированное удаление и обновление записей, хотя и не обеспечивает полноценной защиты.

Стратегии защиты программ ODBC

Ниже перечислены рекомендуемые стратегии защиты программ ODBC.

Ограничение доступа программ к базе данных

Системным администраторам часто приходится ограничивать доступ к определенным файлам, программам или наборам программ. В системе AS/400 это достигается за счет применения принятых прав доступа к программам. Аналогичные методы применимы и при работе с ODBC.

Хранимые процедуры позволяют реализовывать принятые права доступа к программам при работе с ODBC. Программист может запретить пользователям выполнять операции над файлами баз данных

из приложений рабочего стола, таких как Microsoft Access или Lotus 1-2-3. Вместо этого он может разрешить обновление базы данных только своему приложению. Для реализации этого необходимо ограничить доступ пользователей к базе данных с помощью средств защиты на уровне объектов или с помощью пользовательских программ выхода. Приложение должно отправлять запросы данных хранимым процедурам и эти процедуры должны обновлять базу данных.

Ограничение ресурсов CPU, выделяемых пользователям

ODBC существенно упростил процедуру доступа к данным в системе iSeries. Однако в некоторых случаях пользователи могут непреднамеренно создавать запросы, слишком часто обращающиеся к CPU. Это может значительно снизить производительность системы, поскольку операции ODBC выполняются с приоритетом интерактивного задания. В системе iSeries поддерживается **диспетчер запросов**. ODBC может инициализировать создание диспетчера очередей (например, с помощью приложения PC) путем вызова хранимой процедуры. Кроме того, это могут сделать API ODBC с помощью параметра тайм-аута запроса. Наконец, инициализировать диспетчер очередей в задании ODBC может пользовательская программа выхода. Ограничение времени задается в параметре QRYTIMLMT команды CL CHGQRYA. Его можно также указать в файле опций запроса (QAQQINI).

Дополнительная информация приведена в книге *SQL Reference*. Обратитесь к электронной версии этой книги в формате HTML или напечатайте PDF-файл: DB2 Universal Database for iSeries books online.

Дополнительная информация приведена в разделе Администрирование серверов хоста iSeries Access.

Протоколы контроля (отслеживание параметров защиты)

Для отслеживания параметров защиты можно применять несколько протоколов. Протокол событий (QHST) содержит информацию об изменениях в конфигурации защиты системы. Для подробного отслеживания функций защиты следует задать параметр QAUDJRN. Если ему присвоено значение *SECURITY, то будут отслеживаться следующие события:

- Изменение прав доступа к объектам
- Создание, изменение, удаление, просмотр и восстановление пользовательских профайлов
- Изменение владельца объекта
- Изменения в программах (CHGPGM), принимающих профайл владельца
- Изменения системных значений и сетевых атрибутов
- Изменение работы подсистем
- Сброс пароля QSECOFR до значения по умолчанию функцией DST
- Запрос пароля по умолчанию администратора защиты DST
- Изменение атрибутов контроля объекта

Дополнительная информация приведена в книге IBM Security - Reference .

Информация, связанная с защитой ODBC

За подробной информацией по вопросам защиты и помощью в реализации вышеуказанных стратегий обратитесь в службу IBM Consultline (1-800-274-0015). Подробная информация по соответствующим разделам приведена в следующих источниках:

- Администрирование серверов хоста
- IBM Security - Reference 
- Резервное копирование и восстановление 

- DB2 Universal Database для iSeries

Устранение неполадок ODBC

В следующих разделах приведены общие рекомендации по поиску и устранению неполадок ODBC iSeries Access для Windows:

- Инструменты диагностики и анализа производительности ODBC
- Сообщения об ошибках
- Устранение неполадок в соединении с сервером iSeries
- Общие ошибки ODBC
- Сбор информации для IBM Support

Инструменты диагностики и анализа производительности ODBC

В приведенной ниже таблице указаны средства диагностики и повышения производительности ODBC на сервере и клиенте: **Средства клиента**

Трассировка ODBC (SQL.LOG)	Администратор ODBC фирмы Microsoft содержит собственную утилиту трассировки вызовов API ODBC из приложений. Дополнительная информация приведена в разделе Сбор данных трассировки ODBC (SQL.LOG).
Утилиты трассировки ODBC	Существуют и более развитые утилиты трассировки ODBC. С помощью таких платных утилит можно собрать подробные данные трассировки точек входа и выхода вызовов API ODBC. К их числу относятся утилиты Trace Tools (Dr. DeeBee) и SST Trace Plus (Systems Software Technology).
CWBPING	Для выполнения команды CWBPING введите в командной строке MS-DOS команду <code>swbping</code> (IP-адрес или имя системы). Например: <code>swbping testsys1</code> или <code>swbping 127.127.127.1</code> CWBPING выдаст список серверов с информацией об их состоянии. Для просмотра справки по команде CWBPING введите ее в командной строке без параметров. Дополнительная информация о CWBPING приведена в разделе проверка состояния сервера.
CWBCOTRC	Для выполнения трассировки CWBCOTRC введите CWBCOTRC ON в командной строке MS-DOS в каталоге <code>\Program Files\IBM\Client Access</code> . Включив трассировку, можно запустить приложение. Команда CWBCOTRC OFF останавливает трассировку. Программа CWBCOTRC собирает информацию о данных, которыми клиент обменивается с сервером. Для получения справки по команде CWBCOTRC выполните эту команду без параметров.
Подробная трассировка	Подробная трассировка собирает информацию о всех применяемых компонентах iSeries Access для Windows. Эта трассировка содержит следующую информацию об ODBC: точки входа в драйвер, сведения о предварительном задании, имя применяемого пакета и особые ошибки. Дополнительная информация приведена в разделе Сбор данных подробной трассировки.

Средства сервера

Трассировка линий связи	Трассировщик линий связи может выполнять трассировку и форматирование любой линии, для которой существует описание линии (Token-Ring и Ethernet). Этот инструмент позволяет устранить многие неполадки. Он также полезен при поиске "узких мест", снижающих производительность. Поля системного времени и сервера позволяют определить, сколько времени займет обработка запроса.
-------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Трассировка заданий	<p>Трассировка заданий позволяет устранить большую часть неполадок и задержек при выполнении операций. Для трассировки задания необходимо сначала запустить служебное задание. Для этого определите полное имя задания ODBC. Затем из любого сеанса эмуляции 5250 запустите служебное задание для данного задания QZDASOINIT с помощью команды STRSRVJOB. После этого выберите один из двух типов трассировки:</p> <p>Трассировка задания Выполняет трассировку внутренних вызовов серверов хоста. Запустите команду TRCJOB *ON.</p> <p>Отладочная трассировка Применяется для оценки производительности приложения и определения причины какой-либо конкретной неполадки.</p> <p>Команду STRDBG можно применить к активному служебному заданию. Эта команда заносит в протокол задания сеанса отладки информацию обо всех решениях, принимаемых оптимизатором запросов. В протокол заносятся примерные значения продолжительности обработки запросов, применяемые пути доступа, ошибки курсоров и т.п. Команду STRDBG можно запустить из вкладки Диагностика окна настройки DSN в разделе Администрирование ODBC или с помощью следующей команды:</p> <p style="text-align: center;">STRDBG UPDPROD(*YES)</p> <p>В протокол задания ODBC могут быть занесены любые ошибки сервера iSeries. Если задание работает в режиме отладки, то в протокол задания будут дополнительно занесены сведения о производительности.</p>
Инструменты анализа производительности	<p>Набор инструментов, предназначенных для анализа производительности, содержит отчеты и утилиты, позволяющие подробно проанализировать работу приложения. Этот набор инструментов позволяет просматривать информацию об использовании CPU, дисков и виртуальной памяти, а также многие другие показатели работы системы. Несмотря на то, что операционная система содержит средства сбора информации о производительности, для анализа этих результатов понадобится отдельная лицензионная программа Performance Tools/400.</p> <p>Кроме того, вы можете воспользоваться средствами Монитор баз данных и Визуальное представление. Дополнительная информация приведена в электронной справке Навигатора iSeries.</p>
Протокол задания QZDASOINIT	<p>Рекомендуется создать и впоследствии просмотреть протокол задания QZDASOINIT. Этот протокол упрощает поиск и устранение неполадок ODBC.</p> <p>Протокол задания можно создать и найти с помощью источника данных. Для создания протокола нужно отметить опцию Печатать протокол задания при отсоединении на вкладке диагностики. Для того чтобы найти протокол задания, откройте сеанс эмуляции PC5250 и выполните команду WRKSPLF с именем пользовательского профайла iSeries, применяемого в соединении ODBC.</p>
QAQQINI (файл опций запроса)	<p>Файл опций запроса содержит множество опций, которые можно использовать для диагностики, настройки производительности и отладки. Дополнительная информация приведена в документации по базе данных. Этот файл можно задать в источнике данных ODBC (DSN).</p>

Сообщения об ошибках ODBC iSeries Access

При возникновении ошибки драйвер ODBC iSeries Access возвращает SQLSTATE (код ошибки ODBC) и сообщение об ошибке. Драйвер генерирует эту информацию на основе как ошибок, обнаруженных им самим, так и ошибок, о которых сообщает СУБД.

Если ошибка возникла в источнике данных, драйвер ODBC iSeries Access преобразует номер внутренней ошибки в соответствующий код SQLSTATE. В случаях, когда драйвер ODBC iSeries Access и диспетчер драйверов Microsoft обнаруживают одну и ту же ошибку, они генерируют соответствующий код SQLSTATE. Драйвер ODBC iSeries Access формирует сообщение об ошибке по сообщению, полученному от СУБД.

Если ошибка возникла в драйвере ODBC iSeries Access или в диспетчере драйверов Microsoft, то драйвер ODBC iSeries Access возвращает сообщение на основе текстовой информации, связанной с кодом SQLSTATE.

Формат сообщений об ошибках

Ниже приведен формат сообщений об ошибках:

[вендор] [компонент ODBC] [источник данных]
сообщение об ошибке

Префиксы в квадратных скобках ([]) указывают на источник ошибки. В следующей таблице показаны значения префиксов, которые возвращает драйвер ODBC iSeries Access.

При возникновении ошибки в источнике данных префиксы [вендор] и [компонент ODBC] указывают вендора и имя компонента ODBC, получившего эту ошибку из источника данных.

Источник ошибки	Значение
Диспетчер драйверов	[Microsoft] [диспетчер драйверов ODBC] [н/д]
Драйвер ODBC iSeries Access	[IBM] [драйвер ODBC iSeries Access] н/д
Сообщения NLS	[IBM] [драйвер ODBC iSeries Access] Столбец #: Номер сообщения об ошибке NLS Текст сообщения об ошибке NLS
Уровень связи	[IBM] [драйвер ODBC iSeries Access] Не удалось установить соединение. Comm RC=xxxx - (текст сообщения), где xxxx - это номер ошибки в десятичном, а не в шестнадцатеричном формате. После номера ошибки приводится ее описание. Примечание: дополнительная информация об идентификаторах сообщений об ошибках приведена в разделе Коды возврата iSeries Access и в электронном руководстве пользователя iSeries Access для Windows.
DB2 UDB для iSeries	[IBM] [драйвер ODBC iSeries Access] [DB2 UDB] Сообщение об ошибке сервера

Просмотр текста сообщения об ошибке DB2 UDB для iSeries:

Для ошибок, начинающихся с:	Выполните следующую команду OS/400:
-----------------------------	-------------------------------------

SQL	DSPMSGD RANGE(SQLxxxx) MSGF(QSQLMSG)
IWS или PWS	DSPMSGD RANGE(ZZZxxxx) MSGF(QIWS/QIWSMSG), где ZZZ - это IWS или PWS

Дополнительная информация о прочих сообщениях об ошибках ODBC приведена в разделе Общие ошибки ODBC.

Сообщения NLS и сообщения о сбоях соединений приведены в электронном руководстве пользователя iSeries Access для Windows, раздел Службные сообщения, сообщения трассировки и сообщения об ошибках.

Устранение неполадок при подключении к серверу iSeries

Каждое соединение ODBC взаимодействует с одной программой сервера баз данных, запущенной на сервере iSeries. Эта программа называется **программой серверов хоста**. При работе с протоколом TCP/IP применяется программа сервера баз данных **QZDASOINIT**. Как правило, она расположена в подсистеме QSYS, однако системный администратор может изменить ее расположение.

Обычно программа запускается в прозрачном режиме, и от пользователя требуется всего лишь убедиться в том, что все необходимые подсистемы и протоколы запущены и работают правильно. Дополнительная информация по работе с заданиями серверов хоста приведена в разделе Управление серверами хоста iSeries Access для Windows.

Если соединение установить не удастся, то обычно драйвер ODBC выдает соответствующее сообщение об ошибке.

Если ODBC не удастся установить соединение с сервером iSeries, то необходимо выполнить действия по устранению неполадок. Ниже перечислены соответствующие задачи:

- Проверка состояния сервера
- Проверка активности необходимых подсистем
- Проверка активности соответствующих предварительных заданий
- Дополнительные рекомендации при работе с TCP/IP

Проверка состояния сервера: В комплект поставки продукта iSeries Access для Windows входит специальная команда для проверки состояния серверов хоста:

```
CWBPING имя-системы
```

где имя-системы - это имя системы.

Команда должна выдать примерно следующую информацию:

Для отмены запроса CWBPING нажмите CTRL-C или CTRL-BREAK

```
I - Проверка соединения с системой MYSYSTEM...
I - Соединение с приложением сервера установлено: Центральный клиент
I - Соединение с приложением сервера установлено: Сетевой файл
I - Соединение с приложением сервера установлено: Сетевая печать
I - Соединение с приложением сервера установлено: Data Access
I - Соединение с приложением сервера установлено: Очереди данных
I - Соединение с приложением сервера установлено: Удаленная команда
I - Соединение с приложением сервера установлено: Защита
I - Соединение с приложением сервера установлено: DDM
I - Соединение с приложением сервера установлено: Telnet
I - Соединение с приложением сервера установлено: Централизованное управление
I - Проверка соединения с системой MYSYSTEM завершена
```

Примечания:

- Для нормальной работы ODBC должны быть запущены сервер базы данных и сервер защиты.
- Если будет выдано сообщение о том, что для данного соединения настроена поддержка SSL, то с этим соединением могут работать только 32-разрядные приложения. Такие соединения не могут использоваться 64-разрядным драйвером ODBC iSeries Access и 64-разрядным комплексом связи OLE DB iSeries Access. Для подключения 64-разрядного приложения к серверу iSeries нужно отключить протокол SSL для соответствующего соединения.

Проверка активности подсистем: Задания ODBC TCP/IP (QZDASOINIT) будут выполняться в подсистеме QSERVER. Проверьте, запущена ли эта подсистема. Возможно, подсистему QSERVER необходимо запустить вручную. Для этого введите следующую команду:

```
STRSBS QSERVER
```

Для автоматического запуска подсистемы при IPL необходимо изменить программу начальной загрузки (по умолчанию запускается QSYS/QSTRUP), добавив в нее команду STRSBS QSERVER.

Помимо подсистемы QSERVER, должна быть запущена также подсистема QSYSWRK.

Проверка активности предварительных заданий: В поставляемой фирмой IBM подсистеме QSERVER по умолчанию предусматривается применение предварительных заданий с целью повышения производительности при инициализации/запуске заданий. Если в системе настроен запуск предварительных заданий, то для установки соединения НЕОБХОДИМО, чтобы соответствующее предварительное задание было активно. Для соединений TCP/IP применяется следующее предварительное задание:

- QZDASOINIT - Программа сервера

Для того чтобы выяснить, активны ли предварительные задания, введите следующую команду:

```
WRKACTJOB SBS(QSERVER)
```

Должны быть активны следующие задания:

Задание	Пользователь	Тип	----Состояние---
QZDASOINIT	QUSER	PJ	ACTIVE (соединение с сокетом)
QZDASRVSD	QUSER	PJ	ACTIVE (соединение с сокетом)

Предварительные задания будут показаны в выводе команды WRKACTJOB, только если соединение уже активно. Для просмотра предварительных заданий в меню WRKACTJOB нажмите

Дополнительные рекомендации по работе с TCP/IP: Проверьте, запущен ли протокол TCP/IP, с помощью следующей команды:

```
NETSTAT *CNN
```

Примечание: для того чтобы убедиться, что протокол TCP/IP запускается вместе с Навигатором, нужно сначала настроить на сервере протокол TCP/IP, а затем выполнить следующие действия:

1. В Навигаторе выберите сервер —> Сеть.
2. Щелкните правой кнопкой на пункте Конфигурация TCP/IP и выберите Утилиты.
3. Выберите Ping.
4. Укажите имя хоста или IP-адрес и нажмите кнопку Отправить пробный пакет.

Если необходимый протокол не запущен, запустите его командой STRTCP.

Проверьте, запущены ли необходимые демоны, с помощью команды NETSTAT *CNN:

Удаленный адрес	Удаленный порт	Локальный порт	Время простоя	Состояние
*	*	as-cent >	000:09:31	Ожидание
*	*	as-signon	000:09:41	Ожидание
*	*	as-svrmap	002:57:45	Ожидание
*	*	as-data >	002:57:45	Ожидание

Если необходимые демоны не запущены, запустите их командой STRHOSTSVR SERVER(*ALL).

- Проверьте, запущен ли QZDASRVSD - демон сокета ODBC.
 - Демон as-database должен находиться в состоянии Ожидание
 - Для просмотра сообщений об ошибках в протоколе задания демона воспользуйтесь командой WRKJOB QZDASRVSD.
- Проверьте, запущен ли демон QZSOMAPD в подсистеме QSYSWRK.
 - Демон as-svrmap должен находиться в состоянии Ожидание (вывод команды NETSTAT *CNN).
 - Для просмотра сообщений об ошибках в протоколе задания демона воспользуйтесь командой WRKJOB QZSOMAPD.

Для определения номера сокета сервера баз данных PC подключается к сокету сервера номеров портов. Он узнает номер сокета, применяемого демоном as-database. Затем PC подключается к нужному сокету, отслеживаемого демоном файлового сервера, QZDASRVSD. Последний подключает PC к предварительному заданию QZDASOINIT в подсистеме QSERVER. После проверки пользовательского профайла и пароля и копирования профайла в предварительное задание будет запущено задание, аналогичное QZDASOINIT. Если это первое соединение для данного PC, то будут применяться два дополнительных сервера: сервер централизованного управления (для лицензирования) и сервер входа в систему (для проверки ИД и пароля пользователя).

Дополнительная информация о проверке состояния TCP/IP приведена в разделе Общие неполадки TCP/IP.

Общие ошибки ODBC

В следующих разделах приведены общие рекомендации по поиску и устранению часто встречающихся неполадок ODBC iSeries Access для Windows:

- Ошибки SQL
- Ошибки хранимых процедур
- Ошибки вывода и непредвиденные ошибки ODBC

Ошибки SQL:

- SQL0113 - Имя &1 недопустимо.
- SQL0114 - Реляционная база данных &1 отлична от текущего сервера &2
- SQL0204 - MYSYSCONF не найден
- SQL0208 - В таблице результатов отсутствует столбец ORDER BY
- SQL0900 - Приложение не подключено
- SQL0901 - Системная ошибка SQL
- SQL5001 - Не задан спецификатор столбца или таблица &2.
- SQL5016 - Имя объекта &1 не соответствует соглашению о присвоении имен
- SQL0104 - Маркер &1 недопустим. Допустимые маркеры: &2
- SQL7008 Операция &1 в &2 недопустима. Код причины - 3

Примечание: дополнительная информация об ошибках SQL приведена в разделе Коды и сообщения SQL DB2 Universal Database для iSeries.

Ошибки хранимых процедур: Ниже описаны типичные ошибки хранимых процедур:

- SQL0444 - Не найдена внешняя программа &A в &B (SQL DB2 UDB для iSeries)
- Параметры OUTPUT и INPUT_OUTPUT не содержат выходных данных
- SQL0501 - Курсор CRSR000x неоткрыт

SQL0444 - Не найдена внешняя программа &A в &B (SQL DB2 UDB для iSeries): Сообщение SQL0444 выдается при выполнении (возможно, непосредственном), если сервер баз данных обнаружил объявление процедуры, но не смог найти объект программы. Расположение внешней программы должно совпадать с указанным в системных таблицах каталогов. Учтите, что это расположение задается соглашением о присвоении имен и набором по умолчанию, действующими на момент определения процедуры (командой CREATE PROCEDURE), а не ее вызова. Для того чтобы выяснить, какое расположение определено для имени внешней программы хранимой процедуры, выполните запрос в QSYS2.SYSPROCS и просмотрите значение поля "EXTERNAL_NAME".

Параметры OUTPUT и INPUT_OUTPUT не содержат выходных данных: Возможны следующие причины этой ошибки:

- API ODBC **SQLBindParameter** присвоил параметру **fParamType** неверное значение SQL_PARAM_INPUT.
- Вместо CREATE PROCEDURE была использована функция DECLARE PROCEDURE, а расширенная динамическая поддержка отключена.
- В функциях CREATE PROCEDURE или DECLARE PROCEDURE параметр был неправильно объявлен как IN.
- Программа хранимой процедуры возвратила неверное значение параметра.
-

SQL0501 - Курсор CRSR000x не открыт: Для возврата данных при применении встроенного SQL в программах ILE необходимо указывать опцию компиляции ACTGRP(*CALLER), а не опцию по умолчанию *NEW.

Убедитесь в том, что вместо функции выхода программа выполняет функцию возврата.

Если программа хранимой процедуры выполняет функцию выхода вместо функции возврата, присвойте опции **Закреть курсор SQL** значение *ENDACTGRP. Если опции Закреть курсор SQL присвоено значение *ENDMOD, то курсор будет закрыт до получения данных.

Убедитесь также в том, что в команде CREATE PROCEDURE указано правильное число наборов результатов. Это особенно важно, если наборы результатов являются массивами.

Ошибки вывода и непредвиденные ошибки ODBC: Убедитесь в том, что совместимы версии драйвера ODBC iSeries Access и программы сервера баз данных. Убедитесь в наличии всех необходимых сопутствующих PTF ко всем заказанным PTF (они перечислены в файле readme.txt Пакета обслуживания). Если устранить неполадку не удастся, убедитесь в том, что опция предварительной выборки в Источнике данных ODBC отключена. Если применяется API ODBC SQLExtendedFetch или SQLFetchScroll, либо вы затрудняетесь определить, какой именно, отключите опцию предварительной выборки.

Обратите внимание на то, что курсоры набора результатов хранимой процедуры доступны только для пересылки и чтения.

Двоичные или шестнадцатеричные данные вместо символов ASCII

Значение по умолчанию параметра Преобразование не предусматривает преобразование двоичных данных (CCSID 65535) в текст. CCSID прикрепляется к файлам, таблицам и даже отдельным полям (столбцам). Он определяет таблицу для преобразования данных, например из формата EBCDIC в формат ASCII. CCSID 65535 часто соответствует неотформатированным

данным (двоичным или шестнадцатеричным), например изображениям; такие данные не зависят от языка. Если значение Преобразовывать двоичные данные (CCSID 65535) в текст не выбрано, то неотформатированные данные не будут повреждены.

При выборе значения Преобразовывать двоичные данные (CCSID 65535) в текст CCSID, прикрепленный к данным, будет заменен на CCSID задания. Учтите, что это значение может привести к повреждению "настоящих" двоичных данных.

Сбор информации для IBM Support

Когда вы сообщаете о неполадке в службу IBM Support, не забудьте указать подробную информацию (см. ниже) - это необходимо для максимально быстрого устранения неполадки. Для сбора этой информации выполните следующие действия:

Запишите версию OS/400 и уровень совокупного PTF.	<ol style="list-style-type: none"> 1. Введите команду Показать PTF в командной строке эмулятора терминала: DSPPTF 2. Запишите информацию о выпуске OS/400 в формате VxRxMx. 3. Убедитесь, что источник IPL - ##MACH#B. 4. Нажмите F5 для просмотра сведений о PTF. 5. Запишите ИД первого PTF в списке. Он будет показан в формате Tzxxууу, где xx - это год, ууу - дата в юлианском формате, а z - L или C.
Запишите версию драйвера ODBC.	<ol style="list-style-type: none"> 1. На панели задач выберите Пуск -> Программы -> IBM iSeries Access для Windows -> Администрирование ODBC. Примечание: в системе с 64-разрядным процессором и 64-разрядным драйвером ODBC нужно выбрать Администрирование ODBC (64-разрядное). 2. Откройте вкладку Драйверы. 3. Запишите версию драйвера ODBC iSeries Access.
Запишите версию диспетчера драйверов ODBC.	<ol style="list-style-type: none"> 1. На панели задач выберите Пуск -> Программы -> IBM iSeries Access для Windows -> Администрирование ODBC. Примечание: в системе с 64-разрядным процессором и 64-разрядным драйвером ODBC нужно выбрать Администрирование ODBC (64-разрядное). 2. Откройте вкладку О программе. 3. Запишите версию Диспетчера драйверов.
Сбор данных трассировки	Вероятнее всего, в службе поддержки предложат собрать данные следующих трассировок: трассировки ODBC (SQL.LOG), трассировки линии связи (CWBCOTRC) и подробной трассировки. Дополнительная информация о трассировке приведена в разделе Средства диагностики и повышения производительности ODBC.
Запишите дополнительную информацию	Например, приложение PC, описание ошибки и сведения о драйвере ODBC (32-разрядный или 64-разрядный).

Администрирование серверов хоста

В этом разделе приведено краткое описание серверов, работающих в системе iSeries, а также техническая информация о серверах хоста, применяемых продуктом iSeries Access для Windows. Приведенные здесь сведения относятся не ко всем серверам хоста iSeries, а лишь к серверам, обслуживающим iSeries Access для Windows.

Серверы хоста OS/400

Серверы хоста обрабатывают запросы, поступающие от компьютеров-клиентов и устройств, в частности, запросы на запуск приложений, запросы к базе данных, запросы на печать документа и запросы на выполнение резервного копирования и восстановления. Компьютеры iSeries являются многофункциональными серверами, которые могут применяться для различных целей, например, для работы с файлами, базой данных, приложениями, средствами мультимедиа, электронной почтой, а также для печати данных, отправки факсимильных сообщений и работы с беспроводными средствами связи. Если для выполнения каждой задачи применяется отдельный сервер, то достаточно сложно обеспечить правильную, согласованную работу всех серверов. Намного выгоднее приобрести одну систему, которая выполняет функции всех серверов - в этом случае управлять работой сети станет гораздо проще.

Хотя эти серверы предназначены для обслуживания iSeries Access для Windows, ими могут пользоваться и другие клиенты. В этом разделе данные серверы рассматриваются с точки зрения взаимодействия с iSeries Access для Windows.

Добавление и удаление компонента Сервер хоста OS/400

Все серверы OS/400, описываемые в этом разделе, относятся к оптимизированным серверам, поставляемым вместе с основным компонентом операционной системы OS/400. Для работы с Навигатором iSeries нужно установить компонент Сервер хоста.

Если вы не планируете применять продукты семейства iSeries Access для Windows или iSeries NetServer и хотите удалить компонент Сервер хоста OS/400, то предварительно завершите работу подсистем, применяемых серверами хоста. В частности, завершите работу подсистемы QBASE или QCMN (для серверов хоста с поддержкой APPC), подсистемы QSYSWRK и QUSRWRK (для серверов хоста с поддержкой сокетов) и подсистемы QSERVER (для сервера баз данных и файлового сервера). Если во время удаления компонента одна из указанных подсистем будет активна, может возникнуть ошибка.

- **Серверы хоста OS/400**
В этом разделе описаны некоторые серверы хоста, применяемые в iSeries Access для Windows и связанных с ним продуктах. В iSeries Access для Windows список серверов можно упорядочить по типу или по назначению.
- **Работа с серверами хоста**
В этом разделе описана процедура взаимодействия клиента и сервера, а также основные принципы управления этим взаимодействием. Кроме того, в этом разделе перечислены системные значения и подсистемы, связанные с серверами хоста, а также приведена информация о работе с заданиями серверов в системе iSeries.
- **Использование программ выхода**
В этом разделе описана процедура создания и регистрации программы выхода. Кроме того, здесь описаны параметры программы выхода и приведены примеры таких программ.

Серверы хоста OS/400

Приведенная здесь информация относится лишь к тем серверам, которые применяются программой iSeries Access для Windows. В ней описаны не все серверы хоста системы iSeries. К серверам хоста iSeries Access для Windows относятся:

Список серверов хоста iSeries Access для Windows, упорядоченный по их назначению

Список серверов хоста, упорядоченных по их функции в iSeries Access для Windows.

Файловый сервер

Файловый сервер позволяет клиентам сохранять информацию, например, файлы и программы, на сервере iSeries и работать с ней.

Сервер баз данных

Этот сервер применяется функцией Передача данных, ODBC, базой данных Навигатора, API SQL (API DB) и комплексом связи OLE DB iSeries Access для Windows.

Сервер очередей данных

Предоставляет доступ к очередям данных сервера iSeries.

Сервер сетевой печати

Предоставляет поддержку удаленных принтеров и некоторые дополнительные функции управления печатью.

Центральный сервер

Предназначен для управления лицензиями и выполнения других функций управления клиентами.

Сервер обработки удаленных команд и вызовов программ

Позволяет приложениям PC вызывать команды и программы системы OS/400 и возвращает результаты их выполнения клиенту.

Сервер входа в систему

Выполняет функции управления паролями для серверов хоста, поддерживающих сокет.

Сервер номеров портов

Сообщает клиенту, отправившему запрос на установление соединения, текущий номер порта сервера.

Список серверов хоста iSeries Access для Windows, упорядоченный по их назначению

В приведенной ниже таблице перечислены серверы, применяемые различными функциями iSeries Access для Windows.

Функция Client Access	Сервер OS/400
API для работы с базой данных <ul style="list-style-type: none">• SQL• API ODBC	Сервер баз данных
Передача данных	Сервер баз данных
Драйвер ODBC	Сервер баз данных
Работа с Интегрированной файловой системой с помощью Навигатора	Файловый сервер
API для работы с очередями данных	Сервер очередей данных

Функция Client Access	Сервер OS/400
Комплекс связи OLE DB	<ul style="list-style-type: none"> • Сервер очередей данных • Сервер баз данных • Сервер обработки удаленных команд и вызовов распределенных программ • Сервер входа в систему
Управление лицензиями Применяется при запуске приложения, для работы с которым требуется лицензия (Передача данных и Эмуляция 5250)	Центральный сервер
Получение таблицы преобразования Выполняется только при установлении первого соединения, если на клиенте нет необходимых таблиц преобразования	Центральный сервер
Функции для работы с удаленными командами	Сервер обработки удаленных команд и вызовов распределенных программ
Вызов распределенной программы	Сервер обработки удаленных команд и вызовов распределенных программ
Отправка пароля для проверки и замена пароля с истекшим сроком действия (TCP/IP)	Сервер входа в систему
Сервер сетевой печати GUI и API.	Сервер сетевой печати

Дополнительная информация приведена в документе iSeries Access for Windows Servers and Ports Required, APAR II12227 .

Файловый сервер

Файловый сервер позволяет клиентам сохранять информацию на сервере iSeries и работать с ней. Примером такой информации могут служить файлы и программы. Файловый сервер OS/400 обеспечивает доступ к интегрированной файловой системе сервера iSeries. Для работы с файловыми системами клиенты применяют собственный интерфейс, а не пользовательские интерфейсы и API интегрированной файловой системы.

Интегрированная файловая система представляет собой компонент программы OS/400. Она поддерживает функции потокового ввода-вывода и средства управления памятью, применяемые на персональных компьютерах и в операционной системе UNIX. Эта файловая система содержит всю информацию, хранящуюся на сервере iSeries.

Ниже перечислены основные возможности интегрированной файловой системы:

- Поддержка потоковых файлов, содержащих длинные, непрерывные строки данных. Такая строка данных может представлять собой текст документа или элементы изображения. В папках системы iSeries все документы хранятся в виде потоковых файлов. Другим примером потоковых файлов могут служить файлы PC и файлы UNIX. Поддержка потоковых файлов применяется при работе с приложениями типа клиент-сервер.

- Иерархическая структура каталогов позволяет организовать объекты файловой системы в виде дерева. Для доступа к объекту необходимо указать весь путь от корневого каталога к объекту.
- Стандартный интерфейс для работы с потоковыми файлами, файлами базы данных, документами и другими объектами, хранящимися на сервере iSeries, который может применяться как пользователями, так и приложениями.

На сервере iSeries может быть создано несколько файловых систем с одинаковым интерфейсом. Каждая файловая система позволяет пользователям и приложениям работать с определенными сегментами памяти, представленными в виде логических блоков. Такие логические блоки соответствуют файлам, каталогам, библиотекам и объектам.

Список файловых систем iSeries приведен в разделе Введение в интегрированную файловую систему.

Дополнительная информация об интегрированной файловой системе приведена в разделе База данных и файловые системы.

В зависимости от возможностей программы-клиента, файловый сервер OS/400 предоставляет доступ ко всем файловым системам сервера iSeries, либо только к файловой системе QDLS.

В следующем списке перечислены программы, поставляемые вместе с сервером.

Объекты файлового сервера

Имя программы	Библиотека	Тип объекта	Описание
QPWFSERVSO	QSYS	*PGM	Программа сервера
QPWFSERVS2	QSYS	*PGM	Программа сервера
QPWFSERVSD	QSYS	*PGM	Программа демона
QPWFSERV	QSYS	*JOB	Описание задания сервера
QPWFSEVER	QSYS	*CLS	Класс, к которому относятся все задания файлового сервера и сервера баз данных
QPWFSERVSS	QSYS	*PGM	Программа сервера SSL

Сервер баз данных

Сервер баз данных предоставляет клиентам доступ к функциям программы DB2/400. Ниже перечислены некоторые функции этого сервера:

- Поддержка доступа к удаленным функциям SQL
- Доступ к данным через интерфейсы ODBC
- Функции базы данных (в том числе создание и удаление файлов, а также добавление и удаление элементов файлов)
- Функции для получения информации о файлах базы данных, существующих в системе (например, функции составления каталогов SQL)

Кроме того, сервер баз данных позволяет работать с Архитектурой распределенной реляционной базы данных (DRDA). В данном разделе приведена следующая информация о DRDA:

- Пакеты SQL
- Правила именования DRDA
- Рекомендации и ограничения, которые необходимо учесть при работе с DRDA

Дополнительная информация об архитектуре DRDA приведена в разделе Программирование распределенных баз данных.

В следующем списке перечислены программы, поставляемые вместе с сервером.

Программы сервера баз данных

Имя программы	Библиотека	Описание
QZDASOINIT	QSYS	Программа сервера
QZDASON2	QSYS	Программа настройки сокетов
QZDASRVSD	QSYS	Программа демона
QZDASSINIT	QSYS	Программа сервера SSL
Примечание: Сервер баз данных применяет следующие объекты типа *PGM: QZDANDB, QZDAROI, QZDASQL и QZDACMDP.		

Пакеты SQL: Пакеты SQL позволяют связать операторы SQL, указанные в приложении, с реляционной базой данных. Они позволяют повысить производительность приложений, применяющих динамический SQL, за счет многократного использования информации о запросах SQL. Примером приложения, применяющего динамический SQL, может служить сервер баз данных. Он поддерживает применение пакетов для часто используемых запросов SQL, позволяя многократно применять информацию о связывании.

Дополнительная информация по этому вопросу приведена в следующих разделах:

- Имена пакетов SQL
- Удаление пакетов SQL

Имена пакетов SQL: Сервер баз данных можно использовать в качестве шлюза к другим реляционным базам данных, применяющим DRDA. Сервер баз данных автоматически создает один или несколько пакетов SQL в целевой реляционной базе данных. Имена пакетов генерируются в соответствии с атрибутами, применяемыми сервером в этот момент.

Warning: Temporary Level 7 Header:

Имена пакетов в реляционных базах данных, отличных от сервера iSeries: Пакет создается в наборе QSQL400 на сервере приложений в том случае, если реляционная база данных (RDB) не является сервером iSeries. Если RDB является сервером iSeries, то пакет создается в библиотеке QGPL. Если сервер приложений установлен не на сервере Series, то имя пакета будет выглядеть как QZDabcde, где abcde соответствует применяемым опциям синтаксического анализатора. В следующей таблице перечислены опции имен пакетов.

Опции поля имени пакета

Поле	Описание поля	Опции
a	Формат даты	<ul style="list-style-type: none"> • ISO, JIS • USA • EUR • JUL
b	Формат времени	<ul style="list-style-type: none"> • JIS • USA • EUR, ISO

Поле	Описание поля	Опции
c	Управление фиксацией/десятичный ограничитель	<ul style="list-style-type: none"> • *CS/точка • *CS/запятая • *CHG/точка • *CHG/запятая • *RR/точка • *RR/запятая
d	Ограничитель строки	<ul style="list-style-type: none"> • апостроф • кавычка
e	Максимальное число операторов для каждого пакета	<ul style="list-style-type: none"> • 0 - 64 • 1 - 256 • 2 - 512 • 3 - 1024

Имена пакетов в реляционных базах данных - серверах iSeries

Если сервер приложений установлен на сервере Series, то имя пакета будет выглядеть как QZDA**abcdef**, где **abcdef** соответствует применяемым опциям синтаксического анализатора.

Опции поля имени пакета

Поле	Описание поля	Опции
a	Формат даты	<ul style="list-style-type: none"> • ISO, JIS • USA • EUR • JUL • МДГ • ДМГ • ГМД
b	Формат времени и соглашение о присвоении имен	<ul style="list-style-type: none"> • Соглашения о присвоении имен ISO, JIS и SQL • Соглашения о присвоении имен USA и SQL • Соглашения о присвоении имен EUR и SQL • Соглашения о присвоении имен HMS и SQL • Соглашения о присвоении имен ISO, JIS и системное • Соглашения о присвоении имен USA и системное • Соглашения о присвоении имен EUR и системное • Соглашения о присвоении имен HMS и системное

Поле	Описание поля	Опции
c	Уровень фиксации и десятичная точка	<ul style="list-style-type: none"> • *CS/точка • *CS/запятая • *ALL/точка • *ALL/запятая • *CHG/точка • *CHG/запятая • *NONE/точка • *NONE/запятая
d	Ограничитель строки	<ul style="list-style-type: none"> • апостроф • кавычка
e	Число разделов в пакете	<ul style="list-style-type: none"> • 0 - 64 • 1 - 256 • 2 - 512 • 3 - 1024
f	Разделители даты и времени	<ul style="list-style-type: none"> • Старшие биты символа: • '1100'b - Один из форматов ISO для даты • '1101'b - Разделитель даты: запятая • '1110'b - Разделитель даты: точка • '1111'b - Разделитель даты: двоеточие • Младшие биты символа: • '0001'b - Формат ISO для времени • '0010'b - Разделитель времени: запятая • '0011'b - Разделитель времени: точка • '0100'b - Разделитель времени: косая черта • '0101'b - Разделитель времени: тире • '0110'b - Разделитель времени: пробел

Удаление пакетов SQL: Пакеты, применяемые в функциях DRDA, автоматически создаются системой по мере необходимости. Их рекомендуется периодически удалять. Для удаления пакетов служит команда Удалить пакет SQL (DLTSQLPKG).

Удалять пакеты следует только в том случае, если они редко используются. При необходимости удаленный пакет будет создан заново, но это заметно снизит производительность системы.

Правила именования операторов: В приведенной ниже таблице перечислены все правила именования операторов, установленные сервером баз данных.

Правила именования операторов

Оператор	Динамический SQL	Применение расширенного пакета динамического SQL
Локальный	Имя оператора должно быть задано в соответствии с соглашением о присвоении имен в системе iSeries, однако рекомендуется придерживаться формата STMTxxxx Имя курсора должно соответствовать соглашению о присвоении имен в системе iSeries	Имя оператора должно быть задано в соответствии с соглашением о присвоении имен в системе iSeries, однако рекомендуется придерживаться формата STMTxxxx Имя курсора должно соответствовать соглашению о присвоении имен в системе iSeries
DRDA	Имя оператора должно быть задано в формате STMTxxxx Имя курсора должно быть задано в следующем формате: CRSRyuuu для курсоров, не допускающих прокрутку, и SCRSRyuuu для курсоров, допускающих прокрутку, где yuuu совпадает с xxxx.	Имя оператора должно быть задано в формате Sxxxx Имя курсора, не допускающего прокрутку, должно быть задано в формате Суу, где уу совпадает с xxxx и находится в диапазоне от 1 до 15.

Примечания:

1. Правила именования операторов не являются обязательными в локальной системе, поэтому приложения клиента могут применять те же подготовленные операторы, что и приложения системы iSeries, с помощью системного API QSQPRCED.
2. Сервер добавляет пробел в начало всех имен операторов, заданных в формате STMTxxxx. Для того чтобы приложения хоста и клиента могли применять общие операторы, приложение хоста должно добавить пробел в конец имени, заданного в формате STMTxxxx. Сервер не добавляет пробел в конец имени оператора, если его формат отличен от STMTxxxx.

Рекомендации и ограничения, которые следует учесть при работе с DRDA: При подключении к другим реляционным базам данных с помощью DRDA через сервер баз данных следует учесть, что некоторые функции будут недоступны для использования.

В приведенной ниже таблице перечислены функции, на работу с которыми накладываются ограничения при подключении к удаленной системе через сервер баз данных.

Ограничения на функции при работе с DRDA

Функция	Ограничение
Создать пакет Очистить пакет Удалить пакет	Неподдерживаемые функции
Подготовить	Функция расширенной подготовки оператора недоступна при работе с DRDA.
Поддержка расширенного динамического пакета	<ul style="list-style-type: none"> • Доступна только при работе с сервером iSeries, на котором установлена OS/400 версии V2R3 или выше • Имена операторов из пакета должны указываться в формате 'STMTxxxx', где xxxx - номер раздела
Описать признаки параметров	Доступна только при работе с сервером iSeries.

Функция	Ограничение
Блокировка фиксации	Доступна только при работе с сервером iSeries
Уровень фиксации *NONE	Не поддерживается
Уровень фиксации *CHANGE	Поддерживается только при работе с реляционной базой данных, установленной на сервере iSeries. Для работы с другими базами данных требуется уровень фиксации *CS или *ALL.

Сервер очередей данных

Очередь данных - это объект, который применяется приложениями системы iSeries для обмена информацией. В частности, очереди данных могут применяться для обмена информацией между заданиями. С одной очередью данных могут работать несколько заданий системы iSeries.

В программе iSeries Access для Windows предусмотрены API, позволяющие приложениям PC работать с очередями данных точно так же, как это делают приложения системы iSeries. За счет этого приложения системы iSeries могут обмениваться информацией с процессами, выполняющимися на удаленном PC.

В следующем списке перечислены программы, поставляемые вместе с сервером.

Для работы с сервером очередей данных необходима поддержка сокетов

Имя программы	Библиотека	Описание
QZHQSSRV	QSYS	Программа сервера
QZHQSRVD	QSYS	Программа демона

Сервер сетевой печати

Сервер сетевой печати OS/400 предоставляет клиентам расширенные функции для управления ресурсами печати на сервере iSeries. Сервер печати позволяет клиентам выполнять следующие операции над перечисленными объектами:

Буферный файл

Создание, сдвиг указателя, открытие, чтение, запись, закрытие, блокирование, разблокирование, удаление, перемещение, отправка, вызов программы выхода, изменение атрибутов, получение сообщения, ответ на сообщение, просмотр атрибутов, просмотр содержимого

Задание загрузчика

Запуск, завершение работы и просмотр списка объектов

Принтер

Просмотр атрибутов и списка объектов

Очередь вывода

Блокирование, разблокирование, очистка, просмотр содержимого и просмотр атрибутов

Библиотека

Просмотр содержимого

Файл принтера

Просмотр атрибутов, изменение атрибутов и просмотр содержимого

Сервер сетевой печати

Просмотр и изменение атрибутов

В следующем списке перечислены программы, поставляемые вместе с сервером.

Сервер сетевой печати

Имя программы	Библиотека	Описание
QNPSEVS	QSYS	Программа сервера
QNPSEVD	QSYS	Программа демона

Центральный сервер

Центральный сервер предоставляет клиентам следующие функции:

- Управление лицензиями

При получении первого запроса от функции Передача данных или PC5250 для соответствующего пользователя iSeries Access для Windows резервируется лицензия. Сервер продолжает работу до тех пор, пока не истечет время задержки перед освобождением лицензии. Лицензия блокируется до тех пор, пока она не будет освобождена или не будет завершено задание сервера. Для просмотра списка зарезервированных лицензий откройте окно свойств системы iSeries с помощью Навигатора.

- Получение таблиц преобразования

Центральный сервер предоставляет клиентам необходимые таблицы преобразования. Эти таблицы обычно применяются для преобразования данных из ASCII в EBCDIC и наоборот. В запросе на получение таблицы преобразования клиент должен указать исходный и целевой идентификатор набора символов (CCSID), а также таблицу кодовых знаков, которые должны быть преобразованы. В ответ на запрос сервер передаст клиенту правильную таблицу преобразования.

В следующем списке перечислены программы, поставляемые вместе с сервером.

Программы главного сервера

Имя программы	Библиотека	Описание
QZCSRVS	QSYS	Программа сервера
QZCSRVSD	QSYS	Программа демона

Сервер обработки удаленных команд и вызова распределенных программ

Сервер обработки удаленных команд и вызова распределенных программ позволяет удаленным пользователям и приложениям вызывать команды CL и программы сервера iSeries.

Функция обработки удаленных команд позволяет пользователям передать несколько команд на обработку одному заданию. Кроме того, она обеспечивает более надежную проверку пользователей с ограниченными возможностями (пользователей, в профайлах которых указано LMTCPB =*YES).

Функция обработки вызовов распределенных программ позволяет приложениям вызывать программы iSeries и передавать им параметры (входные и выходные). Когда выполнение программы завершается на сервере iSeries, приложению клиента возвращаются значения выходных параметров. Эта функция обеспечивает приложениям прозрачный доступ к ресурсам сервера iSeries - подключение к серверу и преобразование данных будет выполняться автоматически.

В следующем списке перечислены программы, поставляемые вместе с сервером.

Программы сервера обработки удаленных команд и вызовов распределенных программ

Имя программы	Библиотека	Описание
QZRCSRVS	QSYS	Программа сервера
QZRCSRVSD	QSYS	Программа демона

Сервер входа в систему

Сервер входа в систему предоставляет клиентам различные средства защиты. В частности, он запрещает вход в систему пользователям с паролем, срок действия которого истек, проверяет пароль пользовательского профайла и возвращает идентификационные данные пользователя, которые применяются функцией кэширования паролей и функцией Администрирование приложений Навигатора.

В следующем списке перечислены программы, поставляемые вместе с сервером.

Программы сервера входа в систему

Имя программы	Библиотека	Описание
QZSOSIGN	QSYS	Программа сервера
QZSOSGND	QSYS	Программа демона

Сервер номеров портов

Сервер номеров портов позволяет клиентам узнать номер порта требуемой службы (сервера). Он определяет номера портов по таблице служб TCP/IP.

Ниже указана программа, которая поставляется вместе с сервером номеров портов.

Сервер номеров портов

Имя программы	Библиотека	Описание
QZSOSMAPD	QSYS	Программа сервера номеров портов

Работа с серверами хоста OS/400

В этом разделе приведены основные принципы управления заданиями серверов OS/400. В частности, в нем описаны подсистемы, в которых работают серверы, объекты, влияющие на работу серверов, а также способы управления этими ресурсами.

Для работы с серверами, поставляемыми вместе с программой OS/400, обычно не требуется вносить никаких изменений в конфигурацию системы. Все действия по установке и настройке серверов выполняются во время установки программы OS/400. Однако вы можете изменить некоторые параметры управления заданиями серверов, для того чтобы адаптировать их к текущей среде, устранить неполадки, повысить производительность системы или просмотреть информацию о текущих заданиях. Для этого необходимо знать, каким образом некоторые объекты влияют на работу системы и как их можно изменить. Если вы хотите получить более подробную информацию об управлении системой, то перед чтением остальной информации ознакомьтесь с разделом Управление заданиями.

Настройка соединений между клиентом и сервером

В этом разделе приведена информация о том, каким образом устанавливаются и разрываются соединения между клиентами и серверами хоста. Кроме того, в этом разделе приведены номера портов различных серверов, описание демонов серверов и роли, которую они играют при установке соединения.

Подсистемы в OS/400

В этом разделе приведено описание подсистем OS/400, а также автоматических и предварительных заданий.

Системные значения сервера iSeries

В этом разделе приведено описание системных значений, применяемых при взаимодействии клиентов с сервером.

Просмотр списка заданий серверов в системе iSeries

В этом разделе описано, каким образом можно просмотреть список заданий серверов с помощью Навигатора или обычного меню OS/400.

Применение программ EZ Setup и Навигатор для работы с серверами хоста

В этом разделе описано, каким образом можно узнать, активно ли требуемое соединение, и установить соединение, если оно не активно.

Настройка соединений между клиентом и сервером

При установлении соединения между клиентом и сервером выполняются следующие действия:

1. Для инициализации задания сервера с поддержкой сокетов система клиента подключается к серверу через определенный порт.
2. В системе должен быть запущен демон сервера (с помощью команды STRHOSTSVR), обрабатывающий запросы клиентов на соединение. При получении запроса на соединение демон сервера отправляет внутренний запрос на подключение соединения клиента к заданию сервера.
3. Задание сервера представляет собой предварительное или пакетное задание, запущенное во время обработки запроса клиента на подключение. В дальнейшем клиент обменивается информацией с этим заданием сервера. В первую очередь клиент отправляет имя профайла и пароль пользователя.
4. После проверки имени пользовательского профайла и пароля задание сервера переключается на этот пользовательский профайл и устанавливает те атрибуты, которые заданы для этого профайла, в том числе код учета ресурсов и очередь вывода.

Дополнительная информация по этому вопросу приведена в следующих разделах:

- Номера портов серверов хоста
- Запуск серверов хоста
- Завершение работы серверов хоста

Взаимодействие клиента и сервера

При подключении к серверам системы iSeries программа iSeries Access для Windows устанавливает соединение TCP/IP. Для работы с клиентами оптимизированные серверы применяют поддержку сокетов OS/400. Поддержка сокетов OS/400 совместима со стандартом Software Distributions 4.3, разработанным для TCP/IP. Поддержка сокетов поставляется вместе с продуктом 5769-TC1, который устанавливается на сервере iSeries.

За дополнительной информацией об установлении соединений обратитесь к книге Справочник по настройке TCP/IP.

Номера портов для серверов хоста: У каждого типа сервера есть свой собственный демон, который отслеживает поступление клиентских запросов через определенный порт. У этого правила есть несколько исключений. Например, функция передачи данных через сокет использует демон сервера баз данных; сервер сетевых дисков - демон файлового сервера; сервер виртуальной печати - демон сервера сетевой печати. Кроме того, сервер номеров портов следит за указанным портом и разрешает клиентам применять текущий номер порта для работы с определенным сервером.

Номер порта, отслеживаемого каждым демоном сервера, указан в таблице сетевых служб. Например, демон сервера сетевой печати в начальной конфигурации работает с портом 8474, который связан с именем службы 'as-netprt.' Сервер номеров портов отслеживает стандартный порт.

Для сервера номеров портов стандартным является порт 449. Стандартный порт зарезервирован для исключительного применения компонентом OS/400 Host Servers. По этой причине, запись 'as-svrmap' не следует удалять из таблицы сетевых служб.

Номера портов для каждого демона сервера не являются фиксированными; в таблицу сетевых служб можно внести изменения, если для служб требуется задать другие номера портов. Источник изменения таблицы указывается на вкладке Соединения окна свойств системы в Навигаторе iSeries. Однако имена служб изменять нельзя (см. приведенные ниже таблицы). В противном случае демоны сервера не смогут определить номер порта, через который следует работать с запросами клиентов.

При внесении в таблицу новой записи, указывающей другой порт для какой-либо службы, старая запись должна быть удалена. Это предотвратит дублирование имен служб с разными номерами портов и позволит избежать непредсказуемых результатов при запуске демона сервера.

Номера портов для серверов хоста и сервера номеров портов

Этот раздел содержит информацию о номерах портов для оптимизированных серверов и сервера номеров портов, использующих сокет TCP и протокол SSL.

Запуск серверов хоста: Для запуска серверов хоста OS/400 предназначена команда CL STRHOSTSVR. Эта команда запускает все демоны серверов хоста и демона сервера номеров портов. Она также пытается запустить предварительное задание, связанное с данным сервером.

Примечание:

С помощью Навигатора iSeries можно настроить систему таким образом, чтобы при выполнении команды STRTCP автоматически запускались серверы хоста. Во вновь поставляемых системах это происходит автоматически.

Каждому типу сервера хоста соответствует демон сервера. В каждой системе существует один экземпляр сервера номеров портов. Приложение клиента подключается через порт к демону сервера хоста. Демон сервера принимает запрос на соединение и направляет его на обработку заданию сервера.

Параметры команды STRHOSTSVR:

SERVER

***ALL** Запуск всех демонов серверов хоста и демона сервера номеров портов.

***CENTRAL**

Запуск в подсистеме QSYSWRK демона центрального сервера. Имя задания демона - QZSCSRVSD, а имя предварительного задания сервера - QZSCSRVS.

***DATABASE**

Запуск демона сервера баз данных в подсистеме QSERVER. Имя задания демона - QZDASRVSD; имена связанных предварительных заданий сервера - QZDASOINIT, QZDASSINIT и QTFPJTCP.

***DTAQ**

Запуск в подсистеме QSYSWRK демона сервера очереди данных. Имя задания демона - QZHQSRVD, а связанного предварительного задания сервера - QZHQSSRV.

***FILE**

Запуск демона файлового сервера в подсистеме QSERVER. Имя задания демона - QPWFSERVSD; имена связанных предварительных заданий сервера - QPWFSERVSO, QPWSERVSS и QPWFSERVS2.

***NETPRT**

Запуск в подсистеме QSYSWRK демона сервера сетевой печати. Имя задания демона - QNPSEVRD; имена связанных предварительных заданий сервера - QNPSEVRS и QIWWPPJT.

*RMTCMD

Запускает сервер обработки удаленных команд и демон сервера вызова распределенных команд в подсистеме QSYSWRK. Имя задания демона - QZRCSRVS, а связанного предварительного задания сервера - QZRCSRVS.

*SIGNON

Запуск в подсистеме QSYSWRK демона сервера входа в систему. Имя задания демона - QZSOSGND, а связанного предварительного задания сервера - QZSOSIGN.

*SVRMAP

Запуск в подсистеме QSYSWRK демона сервера номеров портов. Имя задания демона - QZSOSMAPD.

Примечание:

Если задание демона запускается в каталоге QSYSWRK, то связанные предварительные задания сервера по умолчанию будут запускаться в каталоге QUSRWRK. Кроме того, предварительные задания сервера баз данных по умолчанию будут запускаться в подсистеме QUSRWRK.

Дополнительный параметр:

RQDPCL

Указывает протоколы, необходимые для запуска демонов серверов хоста.

Отдельные значения:

***ANY** В момент выполнения команды STRHOSTSVR должен быть запущен протокол TCP/IP. Если протокол TCP/IP не запущен, то будет выдано сообщение PWS300D, а демоны серверов хоста запущены не будут. Кроме того, будет выдано диагностическое сообщение (PWS3008).

*NONE

В момент выполнения команды STRHOSTSVR не обязательно должны быть запущены протоколы связи. Сообщения о незапущенных протоколах не выдаются.

*TCP

В момент выполнения команды STRHOSTSVR должен быть запущен протокол TCP/IP. Если протокол TCP/IP не запущен, то будут показаны сообщения PWS3008 и PWS300D, а демоны серверов хоста запущены не будут.

Ниже приведены несколько примеров применения

Пример: STRHOSTSVR: **Пример 1: Запуск всех демонов серверов хоста**

```
STRHOSTSVR(*ALL)
```

Данная команда запускает все демоны серверов и демон сервера номеров портов, если активен хотя бы один из протоколов передачи данных.

Пример 2: Выборочный запуск демонов серверов

```
STRHOSTSVR SERVER(*CENTRAL *SVRMAP) RQDPCL(*NONE)
```

Данная команда запускает демон центрального сервера и демон сервера номеров портов в подсистеме QSYSWRK, даже если не активен ни один из протоколов передачи данных.

Пример 3: Выбор нужного протокола

```
STRHOSTSVR SERVER(*ALL) RQDPCL(*TCP)
```

Если запущен протокол TCP/IP, то данная команда запускает в подсистеме QSYSWRK все демоны серверов хоста и демон сервера номеров портов.

Завершение работы серверов хоста: Для завершения работы серверов хоста OS/400 предназначена команда CL ENHOSTSVR. Эта команда завершает работу всех демонов серверов хоста и демона сервера номеров портов. Если не указан параметр ENDACTCNN и в момент завершения работы демона к серверу подключены какие-либо приложения клиентов, то задание сервера останется активным до разрыва всех соединений. После этого все запросы клиентов на соединение с данным сервером будут отклоняться до тех пор, пока демон сервера не будет запущен вновь.

Завершение работы демона сервера номеров портов не повлияет на активные соединения клиентов с заданиями сервера. Однако пока сервер номеров портов не будет запущен вновь, все запросы клиентов на получение номера порта демона будут отклоняться.

Параметр ENDACTCNN позволяет закрыть активные соединения с серверами *DATABASE и *FILE. При этом завершается работа заданий серверов, обслуживающих эти соединения. Закрыть активные соединения можно только одновременно с завершением работы соответствующих заданий демона. Если указано ключевое слово *DATABASE, то будет завершена работа заданий QZDASOINIT и QZDASSINIT с активными соединениями. Если указано ключевое слово *FILE, то будет завершена работа заданий QPWFSERVSO и QPWFSERVSS с активными соединениями.

Примечание:

Если команда ENHOSTSVR будет выполнена для неактивного демона, то будет выдано диагностическое сообщение. Для завершения работы активных демонов вызовите команду ENHOSTSVR SERVER(*ALL). Если указан параметр *ALL, то диагностическое сообщение не выдается.

Параметры команды ENHOSTSVR:

SERVER

***ALL** Завершает работу активных демонов серверов хоста и демона сервера номеров портов. При использовании этого параметра другие специальные значения указывать не нужно.

***CENTRAL**

Завершает работу демона центрального сервера в подсистеме QSYSWRK.

***DATABASE**

Завершает работу демона сервера баз данных в подсистеме QSERVER.

***DTAQ**

Завершает работу демона сервера очереди данных в подсистеме QSYSWRK.

***FILE** Завершает работу демона файлового сервера в подсистеме QSERVER.

***NETPRT**

Завершает работу демона сервера сетевой печати в подсистеме QSYSWRK.

***RMTCMD**

Завершает работу демона сервера удаленных команд и сервера вызовов распределенных программ в подсистеме QSYSWRK.

***SIGNON**

Завершает работу демона сервера входа в систему в подсистеме QSYSWRK.

***SVRMAP**

Завершает работу демона сервера номеров портов в подсистеме QSYSWRK.

Необязательный параметр

ENDACTCNN

Указывает, следует ли закрывать активные соединения указанных серверов.

Отдельное значение

*NONE

Активные соединения не будут закрываться.

Значения для конкретных серверов

*DATABASE

Буду закрыты активные соединения, обслуживаемые заданиями серверов QZDASOINIT и QZDASSINIT. Работа заданий этих серверов также будет завершена.

***FILE** Буду закрыты активные соединения, обслуживаемые заданиями серверов QPWFSESRV и QPWFSESRVSS. Работа заданий этих серверов также будет завершена.

Ознакомьтесь с некоторыми примерами применения команды ENDDHOSTSRV.

Пример: ENDDHOSTSRV: **Пример 1: Завершение работы всех демонов серверов хоста**

```
ENDDHOSTSRV SERVER(*ALL)
```

Эта команда завершает работу всех демонов серверов и демонов номеров портов.

Пример 2: Выборочное завершение работы демонов

```
ENDDHOSTSRV SERVER(*CENTRAL *SVRMAP)
```

Эта команда завершает работу демона центрального сервера и демона номеров портов.

Пример 3: Выборочное завершение работы демонов и закрытие активных соединений

```
ENDDHOSTSRV SERVER(*CENTRAL *DATABASE) ENDDACTCNN(*DATABASE)
```

Эта команда завершает работу демона центрального сервера в подсистеме QSYSWRK и демона сервера базы данных в подсистеме QSERVER. Кроме того, закрываются активные соединения с сервером *DATABASE, и завершается работа заданий серверов QZDASOINIT и QZDASSINIT, обслуживающих эти соединения.

Подсистемы на сервере iSeries

Ниже перечислены разделы с описанием подсистем, применяемых различными серверами. Кроме того, в них приведена информация о том, как описания подсистем связаны с заданиями серверов.

Описание подсистемы определяет объем и способ передачи данных на обработку в эту подсистему, а также ресурсы, которые выделяются подсистеме для выполнения заданий.

Подсистемы, применяемые заданиями серверов

Работа с автоматическими заданиями

Задания автозапуска выполняют одноразовую или повторяющуюся работу, связанную с конкретной подсистемой. Задания автозапуска, связанные с конкретной подсистемой, автоматически запускаются при каждом запуске подсистемы.

Работа с предварительными заданиями

Подсистемы заданий сервера: Задания сервера могут работать в различных подсистемах, в зависимости от того, что является их основной задачей. Эти подсистемы перечислены ниже:

QSYSWRK

В этой подсистеме выполняются все задания-демоны (за исключением демона файлового сервера и демона сервера баз данных). Демоны файлового сервера и сервера баз данных выполняются в подсистеме QSERVER.

QUSRWRK

В этой подсистеме выполняются задания следующих серверов:

- Сетевая печать
- Удаленный вызов программы/команды
- Главный сервер
- Очередь данных
- Вход в систему
- База данных

QSERVER

В этой подсистеме должны выполняться задания файлового сервера, связанное с ним задание демона и демон сервера баз данных.

Если эта подсистема неактивна, то любые запросы на подключение к файловому серверу или серверу баз данных не выполняются.

Автоматически запускающиеся подсистемы

Подсистема QSYSWRK автоматически запускается при IPL, независимо от значения, заданного для управляющей подсистемы.

При применении программы запуска по умолчанию, поставляемой в комплекте с системой, подсистемы QSERVER и QUSRWRK автоматически запускаются при IPL. Программа запуска системы определяется системным значением QSTRUPPGM, по умолчанию равным QSTRUP QSYS.

Если вы хотите выбрать другую программу запуска системы, укажите эту программу в системном значении QSTRUPPGM. При создании собственной программы запуска за основу можно взять поставляемую с системой программу QSTRUP из библиотеки QSYS.

Примечание: если в системе используется сервер баз данных или файловый сервер, то после изменения программы запуска необходимо убедиться в том, что новая программа запускает подсистему QSERVER.

Начиная с версии V5R1, протокол TCP/IP запускается системой автоматически, не требуя изменения программы запуска системы. При запуске протокола TCP/IP автоматически запускаются серверы хоста. Также при запуске протокола TCP/IP автоматически проверяется, были ли перед запуском серверов хоста запущены подсистемы QUSRWRK и QSERVER.

Если OS/400 выпуска V5R1 (или более позднего выпуска) устанавливается в системе, в которой уже установлена OS/400 более раннего выпуска, и при этом программа запуска старой версии операционной системы была модернизирована для запуска TCP/IP, то после установки новая версия операционной системы будет запускать TCP/IP самостоятельно, а в программе запуска будет возникать ошибка.

С помощью STRTCP, атрибута IPL, можно запретить системе запускать TCP/IP при IPL. Однако делать это не рекомендуется (по умолчанию это значение равно *YES - Запускать TCP/IP).

Работа с автоматическими заданиями: В подсистеме QSERVER предусмотрено автоматическое задание для заданий файлового сервера и сервера баз данных. Если это задание не работает, серверы запустить нельзя. Завершение этого задания не влечет за собой завершения работы подсистем. При возникновении неполадки, связанной с этим заданием, рекомендуется перезапустить подсистему QSERVER.

В подсистеме QSYSWRK предусмотрено автоматическое задание для всех оптимизированных серверов. Это задание отслеживает события, возникающие при выполнении команды STRTCP. Таким способом задания серверов демонов могут динамически определять, когда активизируется протокол TCP/IP. После активизации протокола задания демонов начинают принимать запросы через соответствующие порты. Если автоматическое задание не запущено и во время запуска протокола TCP/IP работают серверы хоста, то необходимо выполнить следующие команды для начала работы с этим протоколом:

1. ENHOSTSVR *ALL
2. STRHOSTSVR *ALL

Имя автоматического задания - QZBSEVTM. Если это задание не активно, его можно запустить с помощью следующей команды:

```
QSYS/SBMJOB CMD(QSYS/CALL PGM(QSYS/QZBSEVTM)) JOB(QZBSEVTM) JOBD(QSYS/QZBSEJBD)
PRTDEV(*USRPRF) OUTQ(*USRPRF) USER(QUSER) PRTTXT(*SYSVAL) SYSLIBL(*SYSVAL)
CURLIB(*CRTDFT) INLLIBL(*JOB) SRTSEQ (*SYSVAL) LANGID(*SYSVAL) CNTRYID(*SYSVAL)
CCSID(*SYSVAL)
```

Примечание: В системе может быть запущен только один экземпляр программы QZBSEVTM.

Работа с предварительными заданиями: Предварительное задание - это пакетное задание, которое запускается до подключения удаленной программы к серверу. В записях предварительных заданий в описании подсистемы указывается программа, класс и пул памяти, с которыми будет работать данное задание после запуска. Кроме того, в записи предварительного задания необходимо указать атрибуты для создания и управления пулом предварительных заданий.

Предварительные задания позволяют повысить быстродействие при подключении клиента к серверу. Определять записи предварительного задания следует внутри подсистемы. Для работы с предварительными заданиями предусмотрены команды Запустить предварительное задание (STRPJ) и Завершить предварительное задание (ENDPJ).

В документации по системе, относящейся к предварительным заданиям (например, в описании команды DSPACTPJ), термином "запрос на запуск программы" обозначаются только запросы на запуск предварительных заданий, даже в том случае, когда предварительное задание запускается при обработке запроса на установление соединения через сокет.

Примечания:

- Предварительные задания могут использоваться многократно, однако в системе не выполняется автоматическая очистка после возвращения предварительного задания в пул. Число раз, которое предварительное задание может использоваться, определяется значением параметра MAXUSE команд CL ADDPJE и CHGPJE. Это означает, что ресурсы, захваченные пользователем при запуске предварительного задания, должны быть очищены до завершения работы с этим заданием. В противном случае следующий пользователь, запустивший задание, получит эти ресурсы в том состоянии, в котором их оставил предыдущий пользователь. Например, если первый пользователь предварительного задания оставит открытым файл, с которым он работал, то следующий пользователь этого задания также сможет работать с данным файлом.

- Некоторые задания серверов по умолчанию запускаются в подсистеме QUSRWRK или QSERVER. С помощью Навигатора вы можете выбрать другую подсистему для запуска серверов.
 1. Для этого дважды щелкните на значке **Навигатор** → **Сеть** → **Серверы** → **iSeries Access**.
 2. Щелкните правой кнопкой мыши на имени сервера, для которого нужно изменить подсистему, и выберите **Свойства**.
 3. Настройте параметры сервера на странице Подсистемы.

Если вы планируете переместить задания из подсистемы по умолчанию в собственную подсистему, то выполните следующие действия:

1. Создайте описание подсистемы.
2. Добавьте предварительные задания с помощью команды ADDPJE. Присвойте параметру STRJOBS значение *YES.

Если это не будет сделано, задания будут выполняться в стандартной подсистеме.

Предварительные задания поддерживаются всеми серверами OS/400, которые допускают подключение через сокет.

К ним относятся:

Сервер сетевой печати

Сервер обработки удаленных команд и вызовов распределенных программ Главный сервер

Сервер баз данных

Защищенный сервер баз данных

Файловый сервер

Защищенный файловый сервер

Сервер очередей данных

Сервер входа в систему (применяется для всех серверов, поддерживающих подключение через сокет)

Ниже перечислены атрибуты записи предварительного задания и начальные значения, которые присваиваются атрибутам различных серверов хоста функцией подключения через сокет.

Описание подсистемы

Подсистема, содержащая записи предварительного задания.

Сервер OS/400	Значение
Сервер сетевой печати	QUSRWRK
Сервер обработки удаленных команд и вызовов программ	QUSRWRK
Центральный сервер	QUSRWRK
База данных	QUSRWRK
Защищенный сервер баз данных	QUSRWRK
Файловый сервер	QSERVER
Защищенный файловый сервер	QSERVER
Сервер очередей данных	QUSRWRK
Сервер входа в систему	QUSRWRK

Имя и библиотека программы

Программа, вызываемая при запуске предварительного задания.

Сервер OS/400	Значение
Сервер сетевой печати	QSYS/QNPSESRVS
Сервер обработки удаленных команд и вызовов программ	QSYS/QZRCSESRVS
Центральный сервер	QSYS/QZSCSESRVS
База данных	QSYS/QZDASOINIT
Защищенный сервер баз данных	QSYS/QZDASSINIT
Файловый сервер	QSYS/QPWFSESRVSO
Защищенный файловый сервер	QSYS/QPWFSESRVSS
Сервер очередей данных	QSYS/QZHQSRRV
Сервер входа в систему	QSYS/QZSOSIGN

Пользовательский профайл

Пользовательский профайл, под управлением которого выполняется задание. Это имя указывается в качестве пользовательского профайла задания. При получении запроса клиента на запуск сервера предварительное задание переключается на пользовательский профайл, от которого был получен запрос.

Сервер OS/400	Значение
Сервер сетевой печати	QUSER
Сервер обработки удаленных команд и вызовов программ	QUSER
Центральный сервер	QUSER
База данных	QUSER
Защищенный сервер баз данных	QUSER
Файловый сервер	QUSER
Защищенный файловый сервер	QUSER
Сервер очередей данных	QUSER
Сервер входа в систему	QUSER

Имя задания

Имя задания, указанное при запуске.

Сервер OS/400	Значение
Сервер сетевой печати	*PGM
Сервер обработки удаленных команд и вызовов программ	*PGM
Центральный сервер	*PGM
База данных	*PGM
Защищенный сервер баз данных	*PGM
Файловый сервер	*PGM
Защищенный файловый сервер	*PGM
Сервер очередей данных	*PGM
Сервер входа в систему	*PGM

Описание задания

Описание предварительного задания. Обратите внимание, что если указано значение *USRPRF, то будет применяться описание задания, заданное для пользовательского профайла, от имени которого выполняется задание. Это означает, что будет применяться описание задания QUSER. Кроме того, будут применяться некоторые атрибуты из описания задания, связанного с пользовательским профайлом, отправившим запрос, в том числе имя принтера и имя очереди вывода.

Сервер OS/400	Значение
Сервер сетевой печати	QSYS/QZBSJOB
Сервер обработки удаленных команд и вызовов программ	QSYS/QZBSJOB
Центральный сервер	QSYS/QZBSJOB
База данных	*USRPRF
Защищенный сервер баз данных	*USRPRF
Файловый сервер	*USRPRF
Защищенный файловый сервер	*USRPRF
Сервер очередей данных	QSYS/QZBSJOB
Сервер входа в систему	QSYS/QZBSJOB

Запуск заданий

Указывает, нужно ли автоматически запускать предварительные задания при запуске подсистемы. По умолчанию во всех записях предварительных заданий значение этого параметра равно *YES. Оно гарантирует, что будут запущены задания серверов. Это связано с тем, что при выполнении команды STRHOSTSVR запускаются все предварительные задания.

Сервер OS/400	Значение
Сервер сетевой печати	*YES
Сервер обработки удаленных команд и вызовов программ	*YES
Центральный сервер	*YES
База данных	*YES
Защищенный сервер баз данных	*YES
Файловый сервер	*YES
Защищенный файловый сервер	*YES
Сервер очередей данных	*YES
Сервер входа в систему	*YES

Начальное число заданий

Число заданий, запускаемых одновременно с подсистемой. Вы можете изменить это значение в соответствии со своими требованиями.

Сервер OS/400	Значение
Сервер сетевой печати	1
Сервер обработки удаленных команд и вызовов программ	1
Центральный сервер	1
База данных	1

Сервер OS/400	Значение
Защищенный сервер баз данных	1
Файловый сервер	1
Защищенный файловый сервер	1
Сервер очередей данных	1
Сервер входа в систему	1

Порог

Минимальное число доступных предварительных заданий, связанных с данной записью предварительного задания. При достижении этого порога автоматически запускаются дополнительные предварительные задания. Это пороговое значение гарантирует наличие определенного числа заданий в пуле.

Сервер OS/400	Значение
Сервер сетевой печати	1
Сервер обработки удаленных команд и вызовов программ	1
Центральный сервер	1
База данных	1
Защищенный сервер баз данных	1
Файловый сервер	1
Защищенный файловый сервер	1
Сервер очередей данных	1
Сервер входа в систему	1

Число дополнительных заданий

Число дополнительных предварительных заданий, запускаемых при достижении порога.

Сервер OS/400	Значение
Сервер сетевой печати	2
Сервер обработки удаленных команд и вызовов программ	2
Центральный сервер	2
База данных	2
Защищенный сервер баз данных	2
Файловый сервер	2
Защищенный файловый сервер	2
Сервер очередей данных	2
Сервер входа в систему	2

Максимальное число заданий

Максимальное число предварительных заданий, которое может быть запущено для этой записи.

Сервер OS/400	Значение
Сервер сетевой печати	*NOMAX

Сервер OS/400	Значение
Сервер обработки удаленных команд и вызовов программ	*NOMAX
Центральный сервер	*NOMAX
База данных	*NOMAX
Защищенный сервер баз данных	*NOMAX
Файловый сервер	*NOMAX
Защищенный файловый сервер	*NOMAX
Сервер очередей данных	*NOMAX
Сервер входа в систему	*NOMAX

Максимальное число пользователей

Максимальное число пользователей задания. Если параметр равен 200, то предварительное задание будет завершено после обработки 200 запросов на запуск сервера.

Примечание:

Сервер баз данных не поддерживает многократное использование предварительных заданий, даже если значение этого параметра больше единицы.

Сервер OS/400	Значение
Сервер сетевой печати	200
Сервер обработки удаленных команд и вызовов программ	1
Центральный сервер	200
База данных	1
Защищенный сервер баз данных	200
Файловый сервер	*NOMAX
Защищенный файловый сервер	*NOMAX
Сервер очередей данных	200
Сервер входа в систему	200

Ожидание задания

Этот параметр указывает, должен ли запрос клиента на подключение дожидаться освобождения задания сервера, если достигнуто максимальное число заданий.

Сервер OS/400	Значение
Сервер сетевой печати	*YES
Сервер обработки удаленных команд и вызовов программ	*YES
Центральный сервер	*YES
База данных	*YES
Защищенный сервер баз данных	*YES
Файловый сервер	*YES
Защищенный файловый сервер	*YES
Сервер очередей данных	*YES

Сервер OS/400	Значение
Сервер входа в систему	*YES

Идентификатор пула

Идентификатор пула подсистем, в котором выполняется предварительное задание.

Сервер OS/400	Значение
Сервер сетевой печати	1
Сервер обработки удаленных команд и вызовов программ	1
Центральный сервер	1
База данных	1
Защищенный сервер баз данных	1
Файловый сервер	1
Защищенный файловый сервер	1
Сервер очередей данных	1
Сервер входа в систему	1

Класс

Имя и библиотека класса, к которому относится предварительное задание.

Сервер OS/400	Значение
Сервер сетевой печати	QGPL/QCASERVER
Сервер обработки удаленных команд и вызовов программ	QGPL/QCASERVER
Центральный сервер	QGPL/QCASERVER
База данных	QSYS/QPWFSEVER
Защищенный сервер баз данных	QSYS/QPWFSEVER
Файловый сервер	QSYS/QPWFSEVER
Защищенный файловый сервер	QSYS/QPWFSEVER
Сервер очередей данных	QGPL/QCASERVER
Сервер входа в систему	QGPL/QCASERVER

Если в параметре Запуск заданий записи предварительного задания указано значение *YES, и для всех остальных параметров оставлены их первоначальные значения, то для каждой записи предварительного задания выполняются следующие действия:

- При запуске подсистемы для каждого сервера запускается одно предварительное задание.
- Для обработки первого запроса клиента на подключение к серверу применяется начальное задание. При этом превышает установленный порог.
- В системе запускаются дополнительные задания сервера, число которых определяется соответствующим параметром записи предварительного задания.
- В системе всегда доступно по крайней мере одно задание сервера.
- Подсистема периодически проверяет число предварительных заданий, готовых к обработке запросов, и завершает лишние задания. При этом число предварительных заданий в подсистеме никогда не становится меньше значения, указанного в параметре Начальное число заданий.

Контроль за работой предварительных заданий

Для контроля за работой предварительных заданий предназначена команда Показать список активных предварительных заданий (DSPACTPJ). Например, для контроля за работой предварительных заданий сервера входа в систему необходимо знать имя подсистемы, в которой выполняются предварительные задания (QUSRWRK или пользовательская подсистема) и имя программы (например, QZSOSIGN).

Ниже приведен пример вывода команды DSPACTPJ:

```
+-----+
                Показать список активных предварительных заданий  AS400597
                                01/12/95  16:39:25
Подсистема . . . . . : QUSRWRK      Дата сброса. . . . . : 01/11/95
Программа. . . . . : QZSOSIGN      Время сброса . . . . . : 16:54:50
Библиотека . . . . . : QSYS        Истекшее время . . . . : 0023:12:21

Предварительные задания:
Текущее число . . . . . : 10
Среднее число . . . . . : 8.5
Максимальное число . . . . . : 25

Применяемые предварительные задания:
Текущее число . . . . . : 5
Среднее число . . . . . : 4.3
Максимальное число . . . . . : 25

                                Еще...
+-----+

                01/12/95  16:39:25
Подсистема . . . . . : QUSRWRK      Дата сброса. . . . . : 01/11/95
Программа. . . . . : QZSOSIGN      Время сброса . . . . . : 16:54:50
Библиотека . . . . . : QSYS        Истекшее время . . . . : 0023:12:21

Запросы на запуск программ:
Текущее число ожидающих запросов . . . . . : 0
Среднее число ожидающих запросов . . . . . : .2
Максимальное число ожидающих запросов . . . . . : 4
Среднее время ожидания . . . . . : 00:00:20.0
Число принятых запросов. . . . . : 0
Число отклоненных запросов . . . . . : 0

                                Конец

Для продолжения нажмите Enter.

F3=Выход  F5=Обновить  F12=Отмена  F13=Статистика сбросов
+-----+
```

Управление предварительными заданиями

Для обновления информации об активных предварительных заданиях в меню Показать список активных предварительных заданий нажмите клавишу (F5). С помощью информации о запросах на запуск программ вы можете определить, нужно ли изменить число доступных предварительных заданий. Если будет указано, что некоторые запросы на запуск программы ожидают освобождения

предварительного задания, то рекомендуется изменить число предварительных заданий с помощью команды Изменить запись предварительного задания (CHGPJE).

Если запросы на запуск программ обрабатываются медленно, выполните некоторые из перечисленных ниже действий:

- Увеличьте пороговое значение
- Увеличьте начальное число заданий (параметр INLJOBS)
- Увеличьте число дополнительных заданий (параметр ADLJOBS)

Необходимо добиться, чтобы для обработки каждого запроса было доступно предварительное задание.

Удаление записей предварительных заданий

Если вы не планируете применять предварительные задания для запуска серверов, выполните следующие действия:

1. Завершите работу предварительных заданий с помощью команды ENDPJ.
Предварительные задания, работа которых была завершена с помощью команды ENDPJ, будут запущены вновь при запуске подсистемы, если в параметре Запуск заданий записи предварительного задания указано значение *YES, либо при вызове команды STRHOSTSVR для данного типа серверов. Если предварительное задание не будет запущено одним из указанных способов, то все запросы на запуск соответствующего сервера будут отклоняться.
2. Удалите записи предварительных заданий из описания подсистемы с помощью команды Удалить запись предварительного задания (RMVPJE).
Команда RMVPJE удаляет записи предварительных заданий из описания подсистемы на постоянной основе. После удаления записи сервер начнет обработку новых запросов.

Записи о выполнении

Записи о выполнении из описания подсистемы применяются при передаче задания демона в подсистему. При вызове команды STRHOSTSVR в описание подсистемы добавляются записи о выполнении для заданий демонов серверов хоста. Эти задания выполняются под управлением пользовательского профайла QUSER. Задания демонов, переданные в подсистему QSYSWRK, помещаются в очередь заданий QSYSNOMAX. Задания демонов, переданные в подсистему QSERVER, помещаются в очередь заданий QPWFSERVER.

Задания серверов выполняются в той же подсистеме, что и соответствующее задание демона. Свойства заданий серверов определяются в их записи предварительного задания. Если предварительные задания серверов не применяются, то задания серверов наследуют свойства соответствующего задания демона.

Ниже описана начальная конфигурация заданий демонов сервера в подсистемах, поставляемых фирмой IBM.

Демон сервера сетевой печати

Подсистема	QSYS/QSYSWRK
Очередь заданий	QSYSNOMAX
Пользователь	QUSER
Запись о выполнении	QNPSEVD
Имя задания	QNPSEVD
Класс	QGPL/QCASERV

Порядковый номер	2538
------------------	------

Демон сервера обработки удаленных команд и вызовов программ

Подсистема	QSYS/QSYSWRK
Очередь заданий	QSYSNOMAX
Пользователь	QUSER
Запись о выполнении	QZRCRVSD
Имя задания	QZRCRVSD
Класс	QGPL/QCASERVER
Порядковый номер	2539

Демон центрального сервера

Подсистема	QSYS/QSYSWRK
Очередь заданий	QSYSNOMAX
Пользователь	QUSER
Запись о выполнении	QZSCRVSD
Имя задания	QZSCRVSD
Класс	QGPL/QCASERVER
Порядковый номер	2536

Демон сервера баз данных

Подсистема	QSYS/QSERVER
Очередь заданий	QPWFSEVER
Пользователь	QUSER
Запись о выполнении	QZDASRVSD
Имя задания	QZDASRVSD
Класс	QSYS/QPWFSEVER
Порядковый номер	600

Демон файлового сервера

Подсистема	QSYS/QSERVER
Очередь заданий	QPWFSEVER
Пользователь	QUSER
Запись о выполнении	QPWFSEVRSD
Имя задания	QPWFSEVRSD
Класс	QSYS/QPWFSEVER
Порядковый номер	200

Демон сервера очередей вывода

Подсистема	QSYS/QSYSWRK
Очередь заданий	QSYSNOMAX
Пользователь	QUSER
Запись о выполнении	QZHQSRVD
Имя задания	QZHQSRVD
Класс	QGPL/QCASERVER
Порядковый номер	2537

Демон сервера входа в систему

Подсистема	QSYS/QSYSWRK
Очередь заданий	QSYSNOMAX
Пользователь	QUSER
Запись о выполнении	QZSOSGND
Имя задания	QZSOSGND
Класс	QGPL/QCASERVER
Порядковый номер	2540

Демон сервера номеров портов

Подсистема	QSYS/QSYSWRK
Очередь заданий	QSYSNOMAX
Пользователь	QUSER
Запись о выполнении	QZSOSMAPD
Имя задания	QZSOSMAPD
Класс	QGPL/QCASERVER
Порядковый номер	2541

Системные значения сервера iSeries

Системные значения представляют один из основных способов управления различными аспектами работы системы. Изменив системные значения, пользователь может настроить необходимую рабочую среду. Примерами системных значений могут служить системная дата и список библиотек.

На сервере iSeries предусмотрен большой набор различных системных значений. Ниже перечислены те из них, которые применяются при взаимодействии клиента и сервера.

QAUDCTL

Управление контролем. С помощью этого системного значения можно включить или выключить контроль на уровне объектов и пользователей. Изменение этого системного значения вступает в силу немедленно.

QAUDENDACN

Действие при ошибке журнала контроля. Данное системное значение задает действие, которое выполняется системой, если при добавлении записи в журнал контроля за действиями операционной системы возникает ошибка. Изменение этого системного значения вступает в силу немедленно.

QAUDFRCLVL

Принудительное сохранение журнала контроля. Данное системное значение задает число

записей журнала контроля за действиями, при достижении которого вся информация из журнала сохраняется во вспомогательной памяти. Изменение этого системного значения вступает в силу немедленно.

QAUDLVL

Уровень контроля за действиями. Изменение этого системного значения вступает в силу немедленно для всех заданий, выполняющихся в системе.

QAUTOVRT

Указывает, должна ли система автоматически создавать виртуальные устройства. Это системное значение применяется функцией удаленного входа в систему дисплейных станций и в сеансах Telnet.

QCCSID

Идентификатор кодированного набора символов, задающий:

- Набор идентификаторов схем кодирования
- Идентификаторы наборов символов
- Идентификаторы кодовых страниц
- Дополнительную информацию о кодировании, необходимую системе для однозначного представления графических символов

Это значение зависит от поддержки языка, установленной в системе. Оно указывает, нужно ли преобразовывать данные перед их отправкой пользователю. Значение по умолчанию равно 65535. Оно означает, что данные преобразовывать не нужно.

QCTLSBSD

Описание управляющей подсистемы

QDSPSGNINF

Указывает, нужно ли показывать меню входа в систему в сеансе эмуляции 5250 (функция рабочей станции, PC5250).

QLANGID

Идентификатор языка, применяемого в системе по умолчанию. Он задает CCSID по умолчанию для задания пользователя, если CCSID задания равен 65535. С помощью CCSID задания по умолчанию клиент и сервер определяют, каким образом должны преобразовываться данные, которыми они обмениваются друг с другом.

QLMTSECOFR

Указывает, может ли пользователь со специальными правами доступа ко всем объектам (*ALLOBJ) или служебными правами доступа (*SERVICE) работать с устройствами. Если это значение равно 1, то для работы с устройством у пользователей со специальными правами доступа *ALLOBJ или *SERVICE дополнительно должны быть права доступа *CHANGE.

Указанные сведения относятся к работе с виртуальными устройствами в сеансе 5250. По умолчанию применяется значение 1. Если требуется, чтобы некоторые пользователи могли входить в систему PC, предоставьте им права доступа к устройству и контроллеру, которые применяются этим PC, либо измените данное значение на 0.

QMAXSIGN

Задаёт максимальное число неудачных попыток входа в систему для локальных и удаленных пользователей. При достижении значения QMAXSIGN система выполняет действие, заданное в системном значении QMAXSGNACN.

Если значение QMAXSGNACN равно 1 (выключить устройство), то значение QMAXSIGN не ограничивает число попыток ввода пароля на PC при установлении соединения.

Потенциально это может привести к несанкционированному доступу на PC. Значение QMAXSGNACN должно равняться 2 или 3.

QMAXSGNACN

Указывает, какое действие должна выполнить система при достижении максимального числа попыток входа в систему на каком-либо устройстве. Допустимы значения 1 (выключить устройство), 2 (отключить пользовательский профайл) и 3 (выключить устройство и отключить пользовательский профайл). Значение по умолчанию равно 3.

QPWDEXPIV

Срок действия пароля (в днях). Изменение этого системного значения вступает в силу немедленно.

QPWDLMTAJC

Указывает, разрешено ли указывать цифры в соседних позициях пароля. Изменение этого системного значения вступает в силу при смене пароля.

QPWDLMTCHR

Указывает, можно ли использовать специальные символы в пароле. Изменение этого системного значения вступает в силу при смене пароля.

QPWDLMTREP

Указывает, может ли пароль содержать повторяющиеся символы. Изменение этого системного значения вступает в силу при смене пароля.

QPWDLVL

Задаёт уровень поддержки паролей в системе. Он определяет допустимую длину пароля на сервере iSeries, тип шифрования паролей, а также то, будут ли удалены из системы пароли клиентов Windows 95/98/ME на сервере AS/400 NetServer. Изменение этого системного значения вступает в силу при следующей IPL.

Внимание! Перед установкой значения, разрешающего применение длинных паролей, вы должны обновить все программы-клиенты до версии, поддерживающей длинные пароли (Express V5R1). В противном случае клиенты версии младше V5R1 не смогут подключаться к серверу iSeries.

QPWDMAXLEN

Максимальное число символов в пароле. Изменение этого системного значения вступает в силу при смене пароля.

QPWDMINLEN

Минимальное число символов в пароле. Изменение этого системного значения вступает в силу при смене пароля.

QPWDPOSDIF

Задаёт расположение символов в пароле. Изменение этого системного значения вступает в силу при смене пароля.

QPWDRQDDGT

Указывает, что пароль обязательно должен содержать цифру. Изменение этого системного значения вступает в силу при смене пароля.

QPWDRQDDIF

Указывает, должен ли новый пароль обязательно отличаться от предыдущего.

QPWDLVDPGM

Имя программы проверки паролей и ее библиотеки в данной системе. Вы можете задать как имя программы, так и имя библиотеки. Изменение этого системного значения вступает в силу при смене пароля.

QRMTSIGN

Указывает способ обработки удаленных запросов на вход в систему. Примером такого запроса может служить процедура открытия сеанса TELNET. Допустимы следующие значения:

- ***FRCSIGNON**: Во всех сеансах удаленного входа в систему должна выполняться обычная процедура входа в систему.
- ***SAMEPRF**: В сеансах удаленного входа в систему дисплейной станции 5250 или рабочей станции процедура входа в систему может не выполняться, если имена исходного и целевого пользовательских профайлов совпадают. Кроме того, процедура входа в систему может не выполняться в сеансах TELNET.
- ***VERIFY**: Процедура входа в систему не выполняется, если у пользователя есть права на вход в систему.
- ***REJECT**: Запрещает удаленный вход в систему дисплейной станции 5250 и удаленный вход для рабочих станций. Однако пользователям разрешено подключаться к системе с помощью TELNET. При этом будет выполняться обычная процедура входа в систему. Для того чтобы запретить подключение к системе с помощью TELNET, завершите работу серверов TELNET.
- **'программа библиотека'**: Пользователь может задать библиотеку (или значение *LIBL) и имя программы, которая будет определять, какие удаленные сеансы разрешено устанавливать, и каким пользователям каких компьютеров разрешен автоматический вход в систему. Эту опцию можно задать только для функции удаленного входа в систему.

Кроме того, это значение задает имя программы, определяющей, какие удаленные сеансы разрешено устанавливать.

Значение по умолчанию равно *FRCSIGNON. Если вы хотите, чтобы пользователи эмулятора 5250 могли устанавливать опцию пропуска меню входа в систему, укажите значение *VERIFY.

QSECURITY

Уровень защиты системы. Изменение этого системного значения вступает в силу при следующей IPL.

- 20 - для входа в систему необходимо указать пароль.
- 30 при входе в систему должен проверяться пароль пользователя, а при каждом обращении к объекту - права доступа. Для работы с любым ресурсом системы у пользователя должны быть соответствующие права доступа.
- 40 - при входе в систему должен проверяться пароль пользователя, а при каждом обращении к объекту - права доступа. Запрещено обращаться к объектам через неподдерживаемые интерфейсы.
- 50 - при входе в систему должен проверяться пароль пользователя, а при каждом обращении к объектам и ресурсам системы - права доступа. На этом уровне обеспечивается защита и целостность библиотеки QTEMP и пользовательских объектов. Программам запрещается обращаться к объектам через неподдерживаемые интерфейсы и передавать поддерживаемым интерфейсам неподдерживаемые значения параметров.

QSTRUPPGM

Программа, которая выполняется при запуске управляющей подсистемы и системы. Эта программа выполняет функции настройки, например, запуск подсистем.

QSYSLIBL

Системная часть списка библиотек. Поиск объектов в первую очередь выполняется в библиотеках из этой части списка. Данный список применяется для поиска объектов некоторыми функциями клиентов.

Просмотр списка заданий серверов в системе iSeries

С помощью эмулятора или обычных меню системы iSeries довольно сложно определить, какое задание связано с некоторым персональным компьютером или функцией клиента. Однако это необходимо знать при анализе неполадок и настройке производительности системы. Список заданий серверов можно просмотреть с помощью Навигатора.

1. Дважды щелкните на значке **Навигатор**.

2. Откройте список **Сеть**, щелкнув на значке **+**.
3. Откройте список **Серверы**, щелкнув на значке **+**.
4. Выберите тип серверов, список заданий которых необходимо просмотреть (например, TCP/IP или iSeries Access для Windows).
5. На правой панели появится список серверов выбранного типа. Щелкните на имени сервера, список заданий которого вы хотите просмотреть, и нажмите кнопку **Задания сервера**. Появится окно со списком заданий сервера, в котором помимо имени задания будет указано имя пользователя, тип задания, состояние задания, а также время и дата его появления в системе.

Ниже перечислены разделы, в которых вы найдете информацию о том, как просмотреть список заданий серверов с помощью обычных меню системы iSeries.

- Подсистемы на сервере iSeries
- Имена заданий сервера iSeries
- Просмотр список заданий серверов
- Просмотр протокола хронологии
- Просмотр списка заданий серверов, запущенных пользователем

Имена заданий сервера iSeries: Имя задания в системе iSeries состоит из трех компонентов:

- Краткое имя задания
- ИД пользователя
- Номер задания (присваиваются по возрастанию)

Имена заданий серверов образуются по следующим правилам:

- Имя задания
 - Для всех заданий, отличных от предварительных, имя задания сервера совпадает с именем программы сервера.
 - Предварительным заданиям присваивается имя, указанное в записи предварительного задания.
 - Заданиям, запущенным серверами, присваивается имя описания задания, а в случае пакетных заданий - имя, указанное при запуске этих заданий (примером сервера, запускающего задания, может являться файловый сервер).
- ИД пользователя
 - Независимо от того, применяются ли предварительные задания, в качестве ИД пользователя всегда указывается QUSER.
 - В протоколе задания указывается имя пользователя, работающего с заданием.
- Номер задания создается автоматически.

Просмотр списка заданий серверов: Существует два способа просмотреть информацию о заданиях серверов. Во-первых, вы можете вызвать команду WRKACTJOB. Во-вторых, можно просмотреть протокол хронологии и узнать, какие задания применяются различными клиентами.

Warning: Temporary Level 6 Header:

Просмотр списка активных заданий с помощью команды WRKACTJOB: Команда WRKACTJOB позволяет просмотреть список активных заданий, включающий демонов серверов и демона сервера номеров портов.

Ниже приведены примеры вывода команды WRKACTJOB. В меню перечислены только те задания, которые относятся к серверам. Для просмотра списка предварительных заданий нажмите **(F14)**.

В примерах меню приведены следующие типы заданий:

Просмотр протокола хронологии: Когда пользователь компьютера-клиента подключается к заданию сервера, это задание переключается на профайл этого пользователя. Для того чтобы узнать, какое задание связано с профайлом пользователя, просмотрите протокол хронологии с помощью команды DSPLOG. Найдите сообщения, начинающиеся с кода:

- CRIAD0B (для сообщений сервера входа в систему)
- CRIAD09 (для сообщений всех остальных серверов)

Просмотр списка заданий серверов, запущенных пользователем: Для просмотра списка заданий серверов, связанных с пользователем, выполните следующие действия:

1. Откройте **Навигатор iSeries** (дважды щелкните на соответствующем значке).
2. Откройте список **Пользователи и группы**, а затем **Все пользователи**.
3. Щелкните правой кнопкой мыши на имени пользователя, для которого вы хотите просмотреть список заданий серверов.
4. Выберите опцию **Объекты пользователя** и нажмите кнопку **Задания**. Появится окно со списком заданий серверов, запущенных этим пользователем.

Для просмотра списка заданий можно вызвать команду WRKOBJLCK. Укажите имя пользовательского профайла и значение *USRPRF.

Применение программ EZ-Setup и Навигатор iSeries для работы с серверами хоста

Программы EZ-Setup и Навигатор iSeries позволяют подключиться к серверу входа в систему, центральному серверу или серверу обработки удаленных команд и вызовов распределенных программ даже тогда, когда на сервере iSeries не запущен протокол связи. Другими словами, программа EZ-Setup может подключиться к серверу до выполнения команды STRTCP. За счет этого программа EZ-Setup может выполнить начальную настройку сервера iSeries до того, как будут настроены и запущены протоколы связи. В этом разделе описано, каким образом можно узнать, активно ли соединение с сервером, применяемое программами EZ-Setup и Навигатор AS/400, и как это соединение установить.

Информация о настройке соединения, применяемого программой EZ-Setup, приведена в электронной справке по этой программе.

Для работы соединения, применяемого программой EZ-Setup, в подсистеме QSYSWRK должны быть запущены три задания QNEOSOEM. В подсистеме QSYSWRK предусмотрено автоматическое задание для активизации этого соединения. Это автоматическое задание, QNEOSOEM, запускает в подсистеме QSYSWRK два других задания с именами QNEOSOEM. Если по каким-то причинам некоторые из этих заданий не активны, запустите их с помощью следующей команды:

```
QSYS/SBMJOB CMD(QSYS/CALL PGM(QSYS/QNEOSOEM)) JOB(QNEOSOEM)
JOBQ(QSYS/QNEOJOBQ) JOBQ(QSYS/QSYSNOMAX) PRTDEV(*JOBQ) OUTQ(*JOBQ)
USER(*JOBQ) PRTTXT(*JOBQ) SYSLIBL(*SYSVAL) INLLIBL(*JOBQ)
LOGCLPGM(*YES) MSGQ(*NONE) SRTSEQ(*SYSVAL) LANGID(*SYSVAL)
CNTRYID(*SYSVAL) CCSID(*SYSVAL)
```

При необходимости эта команда запустит все три задания QNEOSOEM.

Работа с программами выхода сервера

С помощью программ выхода системный администратор может ограничить список операций, которые пользователь клиента может выполнять на сервере. Все серверы поддерживают пользовательские программы выхода. В этом разделе приведена информация о применении и настройке программ выхода. Кроме того, в нем содержатся примеры программ выхода, предназначенных для управления доступом к функциям сервера.

- Регистрация программ выхода
- Создание программ выхода

- Параметры программ выхода
- Примеры программ выхода

Примечание: ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

Регистрация программ выхода

Для того чтобы указать серверу, какую программу выхода он должен применять, зарегистрируйте эту программу выхода. Это можно сделать с помощью средства регистрации OS/400.

Работа со средством регистрации

Для работы со средством регистрации предназначена команда Работа с информацией о регистрации (WRKREGINF).

```

+-----+
|                                     |
|                   Работа с информацией о регистрации (WRKREGINF)           |
|                                     |
| Введите варианты, нажмите Enter.                                         |
|                                     |
| Точка выхода . . . . . *REGISTERED                                       |
| Формат точки выхода. . . . . *ALL      Имя, шаблон*, *ALL                 |
| Вывод . . . . . *      *, *PRINT                                         |
|                                     |
+-----+

```

Для просмотра списка зарегистрированных точек выхода нажмите Enter.

```

+-----+
|                                     |
|                   Работа с информацией о регистрации                       |
|                                     |
| Введите опции, нажмите Enter.                                             |
| 5=Показать точку выхода   8=Работа с программами выхода                 |
|                                     |
|      Точка                               Формат                             |
| Опц  выхода                             выхода  Регистрация  Описание     |
| _    QIBM_QGW_NJEOUTBOUND  NJE00100   *YES      Запись сетевого задания |
| 8    QIBM_QHQ_DTAQ         DTAQ0100   *YES      Сервер очередей данных  |
| _    QIBM_QLZP_LICENSE     LICM0100   *YES      Управление лицензиями |
| _    QIBM_QMF_MESSAGE      MESS0100   *YES      Сервер сообщений       |
| _    QIBM_QNPS_ENTRY       ENTR0100   *YES      Сервер сетевой печати - |
| _    QIBM_QNPS_SPLF        SPLF0100   *YES      Сервер сетевой печати - |
| _    QIBM_QNS_CRADDACT     ADDA0100   *YES      Добавить описание CRQ  |
| _    QIBM_QNS_CRCHGACT     CHGA0100   *YES      Изменить описание CRQ  |
| _    QIBM_QNS_CRDLTSBMCRQ  DLTA0100   *YES      Удалить запущенный CRQ |
| _    QIBM_QNS_CRDSPACT     DSPA0100   *YES      Показать описание CRQ  |
| _    QIBM_QNS_CREXCACT     EXCA0100   *YES      Выполнить функции CRQ  |
|                                     |
| Команда                               |
| ===>                                  |
|                                     |
+-----+

```

Для работы с программами выхода, связанными с точкой выхода, определенной для сервера, выберите опцию 8.

```

+-----+
|                                     |
|                   Работа с программами выхода                             |
|                                     |
| Точка выхода:   QIBM_QHQ_DTAQ           Формат:   DTAQ0100             |
|                                     |
| Введите опции, нажмите Enter.                                         |
|                                     |
+-----+

```

```

1=Добавить   4=Удалить   5=Показать   10=Заменить

      Номер
      программы
Опц   выхода   Программа   Библиотека
1_   _____
      (Программы выхода не найдены)

```

Для добавления программы выхода, связанной с точкой выхода, выберите опцию 1.

Примечания:

- Если для точки выхода уже определена программа выхода, то ее необходимо удалить перед добавлением новой программы.
- Хотя средство регистрации позволяет определить несколько пользовательских программ выхода для одной точки выхода, серверы всегда применяют программу выхода, указанную в списке первой.
- Для того чтобы изменения вступили в силу, необходимо перезапустить предварительные задания.

```

-----+-----
                          Добавить программу выхода (ADDEXITPGM)
Введите варианты, нажмите Enter.
Точка выхода . . . . . > QIBM_QHQ_DTAQ
Формат точки выхода. . . . . > DTAQ0100      Имя
Номер программы. . . . . > 1                  1-2147483647, *LOW, *HIGH
Программа. . . . . MYPGM                      Имя
Библиотека . . . . . MYLIB Имя, *CURLIB
'Описание' . . . . . *BLANK

```

Укажите библиотеку и имя программы, связанной с указанной точкой выхода.

Одну и ту же программу можно задать для нескольких точек выхода. Такая программа может обрабатывать различные типы запросов, основываясь на полученных входных данных.

Ниже перечислены точки выхода и их форматы для различных серверов OS/400.

QIBM_QPWFS_FILE_SERV (Файловый сервер)

Имя формата	PWFS0100
Имя приложения	*FILESRV

QIBM_QZDA_INIT (Инициализация сервера баз данных)

Имя формата	ZDAI0100
Имя приложения	*SQL

QIBM_QZDA_NDB1 (Внутренние запросы к серверу баз данных)

Имена форматов	ZDAD0100 ZDAD0200
Имя приложения	*NDB

QIBM_QZDA_SQL1 (Запросы SQL к серверу баз данных)

Имена форматов	ZDAQ0100 ZDAQ0200
Имя приложения	*SQLSRV

QIBM_QZDA_ROI1 (Запросы к серверу баз данных для получения информации об объекте)

Имена форматов	ZDAR0100 ZDAR0200
Имя приложения	*RTVOBJINF

QIBM_QZHQ_DATA_QUEUE (Сервер очередей данных)

Имя формата	ZHQ00100
Имя приложения	*DATAQSRV

QIBM_QNPS_ENTRY (Сервер сетевой печати)

Имя формата	ENTR0100
Имя приложения	QNPSERVER

QIBM_QNPS_SPLF (Сервер сетевой печати)

Имя формата	SPLF0100
Имя приложения	QNPSERVER

QIBM_QZSC_LM (Запросы на управление лицензиями к центральному серверу)

Имя формата	ZSCL0100
Имя приложения	*CNTRLSRV

QIBM_QZSC_NLS (Запросы NLS к центральному серверу)

Имя формата	ZSCN0100
Имя приложения	*CNTRLSRV

QIBM_QZRC_RMT (Сервер обработки удаленных команд и вызовов распределенных программ)

Имя формата	CZRC0100
Имя приложения	*RMTSRV

QIBM_QZSO_SIGNONSRV (Сервер входа в систему)

Имя формата	ZSOY0100
Имя приложения	*Signon

Создание программ выхода

Если для сервера определена программа выхода, то перед обработкой запроса он передает этой программе следующие два параметра:

- Код возврата размером в один байт
- Структуру, содержащую информацию о запросе (каждой точке выхода передается своя структура данных).

Если для сервера определена программа выхода, то перед обработкой запроса он передает этой программе следующие два параметра.

- Код возврата размером в один байт
- Структуру, содержащую информацию о запросе (каждой точке выхода передается своя структура данных).

С помощью полученных параметров программа выхода определяет, можно ли обработать запрос. Если программа выхода устанавливает код возврата X'F1', сервер разрешает обработку запроса. Если программа выхода устанавливает код возврата X'F0', сервер отклоняет запрос. Действия, выполняемые сервером при получении кода возврата, отличного от X'F1' и X'F0', зависят от типа этого сервера.

Одна и та же программа выхода может быть определена для нескольких серверов и точек выхода. В этом случае с помощью второго входного параметра программа определяет тип сервера, которому был отправлен запрос, и имя требуемой функции.

Формат второго параметра программ выхода описан в разделе Формат параметров программ выхода. Эта информация может пригодиться при создании собственных программ выхода.

Параметры программ выхода

В перечисленных ниже разделах описаны структуры данных, которые передаются программам для точек выхода серверов OS/400 в качестве второго параметра.

- Файловый сервер
- Сервер баз данных
- Сервер очередей данных
- Сервер сетевой печати
- Центральный сервер
- Сервер обработки удаленных команд и вызовов распределенных программ
- Сервер входа в систему

Файловый сервер: Для файлового сервера определена одна точка выхода:

QIBM_QPWFS_FILE_SERV в формате PWFS0100

Точка выхода QIBM_QPWFS_FILE_SERV предназначена для запуска программы выхода при получении файловым сервером запросов следующих типов:

- Изменить атрибуты файла
- Создать потоковый файл или каталог
- Удалить файл или каталог
- Показать атрибуты файла
- Переместить
- Открыть потоковый файл
- Переименовать
- Выделить диалог

Примечание:

Имя программы выхода файлового сервера определяется при активации подсистемы QSERVER. В связи с этим после изменения имени программы необходимо перезапустить указанную подсистему.

Точка выхода QIBM_QPWFS_FILE_SERV, формат PWFS0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для файлового сервера это значение равно *FILESRV.
20	14	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена: <ul style="list-style-type: none"> • X'0000' - Изменить атрибуты файла • X'0001' - Создать потоковый файл или каталог • X'0002' - Удалить файл или каталог • X'0003' - Показать атрибуты файла • X'0004' - Переместить • X'0005' - Открыть потоковый файл • X'0006' - Переименовать • X'0007' - Выделить диалог
24	18	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QPWFS_FILE_SERV имя формата равно PWFS0100.
32	20	CHAR(4)	Доступ к файлу	Если в поле функции указано значение '5' (открыть), то в данном поле содержится следующая структура: <ul style="list-style-type: none"> • Доступ на чтение, CHAR(1) X'F1' - Да X'F0' - Нет • Доступ на запись, CHAR(1) X'F1' - Да X'F0' - Нет • Доступ на чтение и запись, CHAR(1) X'F1' - Да X'F0' - Нет • Разрешено удаление объекта, CHAR(1) X'F1' - Да X'F0' - Нет
36	24	BINARY(4)	Длина имени файла	Длина имени файла, указанного в следующем поле. Максимальная длина составляет 16 Мб.
40	28	CHAR(*)	Имя файла	Имя файла. Длина этого поля задается в поле Длина имени файла (в предыдущем поле). Имя файла возвращается в кодировке ISO/IEC 10646 (UCS—2, уровень 1), CCSID 61952.

Примечание:

- Определение этого формата содержится в элементе EPWFSEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLESRC, расположенных в библиотеке QSYSINC.
- Дополнительная информация о наборе символов ISO/IEC 10646 (UCS—2, уровень 1) приведена в документе *Information Standard, ISO/IEC 10646—1: Information technology — Universal—Octet Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane*, номер ISO/IEC 10646—1: 1993(E).

Для преобразования данных из набора символов UCS—2, уровень 1, и в него предусмотрены API iconv() и CDRCVRT.

Сервер баз данных: Для сервера баз данных определены четыре точки выхода:

1. QIBM_QZDA_INIT
 - Вызывается при инициализации сервера
2. QIBM_QZDA_NDB1
 - Вызывается при получении внутренних запросов к базе данных
3. QIBM_QZDA_SQL1
 - Вызывается при получении запросов SQL
4. QIBM_QZDA_SQL2
 - Вызывается при получении запросов SQL
5. QIBM_QZDA_ROI1
 - Вызывается для обработки запросов на получение информации об объекте и запросов на вызов функций из каталога SQL

Для точек выхода, применяемых для обработки внутренних запросов к базе данных и запросов на получение информации об объекте, определено два формата, зависящих от типа запрошенной функции.

Точка выхода QIBM_QZDA_INIT применяется для вызова программы выхода во время инициализации сервера. Если для этой точки выхода определена программа, то она вызывается при инициализации сервера баз данных.

Точка выхода QIBM_QZDA_INIT, формат ZDAI0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для данной точки выхода применяется идентификатор *SQL.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZDA_INIT имя формата равно ZDAI0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена Для данной точки выхода допустимо только значение 0.
<p>Примечание: Определение этого формата содержится в элементе EZDAEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLSRC, расположенных в библиотеке QSYSINC.</p>				

Точка выхода QIBM_QZDA_NDB1 предназначена для запуска программы выхода при получении внутренних запросов к серверу баз данных. Эта точка выхода поддерживает два формата. Формат ZDAD0100 применяется для следующих функций:

- Создать исходный физический файл
- Создать файл базы данных на основе существующего
- Добавить, очистить или удалить элемент файла базы данных
- Переопределить файл базы данных
- Отменить переопределение файла базы данных
- Удалить файл

Формат ZDAD0200 применяется для обработки запросов на добавление библиотек в список.

Точка выхода QIBM_QZDA_NDB1, формат ZDAD0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для данной точки выхода это значение равно *NDB.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода Для перечисленных ниже функций применяется формат ZDAD0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена Данное поле содержит одно из следующих значений: <ul style="list-style-type: none"> • X'1800' - Создать исходный физический файл • X'1801' - Создать файл базы данных • X'1802' - Добавить элемент к файлу базы данных • X'1803' - Очистить элемент файла базы данных • X'1804' - Удалить элемент файла базы данных • X'1805' - Переопределить файл базы данных • X'1806' - Отменить переопределение файла базы данных • X'1807' - Создать файл сохранения • X'1808' - Очистить файл сохранения • X'1809' - Удалить файл
32	20	CHAR(128)	Имя файла	Имя файла, над которым должна быть выполнена операция
160	A0	CHAR(10)	Имя библиотеки	Имя библиотеки, содержащей файл
170	AA	CHAR(10)	Имя элемента	Имя элемента, который должен быть добавлен, очищен или удален
180	B4	CHAR(10)	Права доступа	Права доступа к создаваемому файлу
190	BE	CHAR(128)	Имя исходного файла	Имя файла, на основе которого должен быть создан новый файл
318	13E	CHAR(10)	Библиотека исходного файла	Имя библиотеки, содержащей исходный файл
328	148	CHAR(10)	Имя переопределяемого файла	Имя файла, который должен быть переопределен
338	152	CHAR(10)	Библиотека переопределяемого файла	Имя библиотеки, содержащей файл, который должен быть переопределен
348	15C	CHAR(10)	Имя переопределяемого элемента	Имя элемента, который должен быть переопределен

Смещение		Тип	Поле	Описание
Дес	Шест			
Примечание: Определение этого формата содержится в элементе EZDAEP файлов H, QRPGRSRC, QRPGLSRC, QLBSLRC и QCBLLSRC, расположенных в библиотеке QSYSINC.				

Точка выхода QIBM_QZDA_NDB1, формат ZDAD0200

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для данной точки выхода это значение равно *NDB.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для функции добавления библиотек в список имя формата равно ZDAD0200.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена X'180C' - Добавить в список библиотек
32	20	BINARY(4)	Число библиотек	Число библиотек, указанных в следующем поле
36	24	CHAR(10)	Имя библиотеки	Имена библиотек
Примечание: Определение этого формата содержится в элементе EZDAEP файлов H, QRPGRSRC, QRPGLSRC, QLBSLRC и QCBLLSRC, расположенных в библиотеке QSYSINC.				

Точка выхода QIBM_QZDA_SQL1 предназначена для запуска программы выхода при получении сервером баз данных некоторых запросов SQL. Для этой точки выхода определен только один формат. Ниже перечислены функции, при обращении к которым вызывается программа выхода:

- Подготовить
- Открыть
- Выполнить
- Подключиться
- Создать пакет
- Очистить пакет
- Удалить пакет
- Выборка из потока
- Немедленно выполнить
- Подготовить и создать описание
- Подготовить и выполнить или подготовить и открыть
- Открыть и выбрать
- Выполнить или открыть

Точка выхода QIBM_QZDA_SQL1, формат ZDAQ0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для данной точки выхода это значение равно *SQLSRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZDA_SQL1 имя формата равно ZDAQ0100.
28	1C	BINARY(4)	Запрошенная функция	<p>Функция, которая будет выполнена</p> <p>Данное поле содержит одно из следующих значений:</p> <ul style="list-style-type: none"> • X'1800' - Подготовить • X'1803' - Подготовить и создать описание • X'1804' - Открыть/создать описание • X'1805' - Выполнить • X'1806' - Немедленно выполнить • X'1809' - Подключиться • X'180C' - Выборка из потока • X'180D' - Подготовить и выполнить • X'180E' - Открыть и выбрать • X'180F' - Создать пакет • X'1810' - Очистить пакет • X'1811' - Удалить пакет • X'1812' - Выполнить или открыть
32	20	CHAR(18)	Имя оператора	Имя оператора, который должен быть подготовлен или выполнен
50	32	CHAR(18)	Имя курсора	Имя курсора, который должен быть открыт
68	44	CHAR(2)	Опция подготовки	Опция, применяемая при подготовке оператора
70	46	CHAR(2)	Атрибуты операции открытия	Параметры, применяемые функцией открытия
72	48	CHAR(10)	Имя расширенного динамического пакета	Имя расширенного пакета динамического SQL
82	52	CHAR(10)	Библиотека пакета	Имя библиотеки, в которой расположен расширенный пакет динамического SQL
92	5C	BINARY(2)	Индикатор DRDA	<ul style="list-style-type: none"> • 0 - Соединение с локальной RDB • 1 - Соединение с удаленной RDB
94	5E	CHAR(1)	Уровень управления фиксацией	<ul style="list-style-type: none"> • 'A' - Фиксация *ALL • 'C' - Фиксация *CHANGE • 'N' - Фиксация *NONE • 'S' - Фиксация *CS (измененные и текущая)

Смещение		Тип	Поле	Описание
Дес	Шест			
95	5F	CHAR(512)	Первые 512 байт текста оператора SQL	Первые 512 байт текста оператора SQL
Примечание: Определение этого формата содержится в элементе EZDAEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLSRC, расположенных в библиотеке QSYSINC.				

Точка выхода QIBM_QZDA_SQL2 применяется для запуска программы выхода при получении сервером баз данных некоторых запросов SQL. Приоритет точки выхода QIBM_QZDA_SQL2 выше, чем у точки выхода QIBM_QZDA_SQL1. Если для точки выхода QIBM_QZDA_SQL2 зарегистрирована программа, то она будет вызываться вместо программы, определенной для точки выхода QIBM_QZDA_SQL1. Ниже перечислены функции, при обращении к которым вызывается программа выхода:

- Подготовить
- Открыть
- Выполнить
- Подключиться
- Создать пакет
- Очистить пакет
- Удалить пакет
- Выборка из потока
- Немедленно выполнить
- Подготовить и создать описание
- Подготовить и выполнить или подготовить и открыть
- Открыть и выбрать
- Выполнить или открыть

Таблица А-6. Точка выхода QIBM_QZDA_SQL2, формат ZDAQ0200

0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для данной точки выхода это значение равно *SQLSRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZDA_SQL1 имя формата равно ZDAQ0100.

28	1C	BINARY(4)	Запрошенная функция	<p>Функция, которая будет выполнена</p> <p>Данное поле содержит одно из следующих значений:</p> <ul style="list-style-type: none"> • X'1800' - Подготовить • X'1803' - Подготовить и создать описание • X'1804' - Открыть/создать описание • X'1805' - Выполнить • X'1806' - Немедленно выполнить • X'1809' - Подключиться • X'180C' - Выборка из потока • X'180D' - Подготовить и выполнить • X'180E' - Открыть и выбрать • X'180F' - Создать пакет • X'1810' - Очистить пакет • X'1811' - Удалить пакет • X'1812' - Выполнить или открыть
32	20	CHAR(18)	Имя оператора	Имя оператора, который должен быть подготовлен или выполнен
50	32	CHAR(18)	Имя курсора	Имя курсора, который должен быть открыт
68	44	CHAR(2)	Опция подготовки	Опция, применяемая при подготовке оператора
70	46	CHAR(2)	Атрибуты операции открытия	Параметры, применяемые функцией открытия
72	48	CHAR(10)	Имя расширенного динамического пакета	Имя расширенного пакета динамического SQL
82	52	CHAR(10)	Библиотека пакета	Имя библиотеки, в которой расположен расширенный пакет динамического SQL
92	5C	BINARY(2)	Индикатор DRDA	<ul style="list-style-type: none"> • 0 - Соединение с локальной RDB • 1 - Соединение с удаленной RDB
94	5E	CHAR(1)	Уровень управления фиксацией	<ul style="list-style-type: none"> • 'A' - Фиксация *ALL • 'C' - Фиксация *CHANGE • 'N' - Фиксация *NONE • 'S' - Фиксация *CS (измененные и текущая)
95	5F	CHAR(10)	Набор SQL по умолчанию	Имя набора SQL, который по умолчанию применяется сервером баз данных системы iSeries
105	69	CHAR(129)	Зарезервировано	Зарезервировано для параметра, который будет добавлен в будущем
234	EA	BINARY(4)	Длина текста оператора SQL	Длина текста оператора SQL в следующем поле. Максимальное значение составляет 32 Кб.
238	EE	CHAR(*)	Текст оператора SQL	Полный текст оператора SQL

Примечание:

Определение этого формата содержится в элементе EZDAEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLSRC, расположенных в библиотеке QSYSINC.

Точка выхода QIBM_QZDA_ROI1 предназначена для запуска программы выхода, обрабатывающей запросы на получение информации об объектах, отправляемые серверу баз данных. Кроме того, она применяется при вызове функций из каталога SQL.

Для данной точки выхода определено два формата. Они описаны ниже.

Формат ZDAR0100 применяется для обработки запросов на получение информации о следующих объектах:

- Библиотека (или набор)
- Файл (или таблица)
- Поле (или столбец)
- Индекс
- Реляционная база данных (или RDB)
- Пакет SQL
- Оператор из пакета SQL
- Элемент файла
- Формат записи
- Столбцы с особой информацией

Формат ZDAR0200 применяется для обработки запросов на получение информации о следующих объектах:

- Внешние ключи
- Первичные ключи

Точка выхода QIBM_QZDA_ROI1, формат ZDAR0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для сервера баз данных это значение равно *RTVOBJINF.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для перечисленных ниже функций применяется формат ZDAR0100.

Смещение		Тип	Поле	Описание
Дес	Шест			
28	1C	BINARY(4)	Запрошенная функция	<p>Функция, которая будет выполнена</p> <p>Данное поле содержит одно из следующих значений:</p> <ul style="list-style-type: none"> • X'1800' - Получить информацию о библиотеке • X'1801' - Получить информацию о реляционной базе данных • X'1802' - Получить информацию о пакете SQL • X'1803' - Получить оператор из пакета SQL • X'1804' - Получить информацию о файле • X'1805' - Получить информацию об элементе файла • X'1806' - Получить информацию о формате записи • X'1807' - Получить информацию о поле • X'1808' - Получить информацию об индексе • X'180B' - Получить информацию об особых столбцах
32	20	CHAR(20)	Имя библиотеки	Имя библиотеки или шаблон для поиска, применяемый для получения информации о библиотеках, пакетах, операторах пакетов, файлах, элементах, формате записей, полях, индексах и особых столбцах.
52	34	CHAR(36)	Имя реляционной базы данных	Имя реляционной базы данных или шаблон для поиска, применяемый для получения информации о RDB
88	58	CHAR(20)	Имя пакета	Имя пакета или шаблон для поиска, применяемый для получения информации о пакете или операторе пакета
108	6C	CHAR(256)	Имя файла (псевдоним SQL)	Имя файла или шаблон для поиска, применяемый для получения информации о файле, элементе, формате записи, поле, индексе или особом столбце
364	16C	CHAR(20)	Имя элемента	Имя элемента или шаблон для поиска, применяемый для получения информации об элементе файла
384	180	CHAR(20)	Имя формата	Имя формата или шаблон для поиска, применяемый для получения информации о формате записей
<p>Примечание: Определение этого формата содержится в элементе EZDAEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLSRC, расположенных в библиотеке QSYSINC.</p>				

Точка выхода QIBM_QZDA_ROI1, формат ZDAR0200

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу

Смещение		Тип	Поле	Описание
Дес	Шест			
10	A	CHAR(10)	Идентификатор сервера	Для сервера баз данных это значение равно *RTVOBJINF.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для перечисленных ниже функций применяется формат ZDAR0200.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена Данное поле содержит одно из следующих значений: <ul style="list-style-type: none"> • X'1809' - Получить информацию о внешнем ключе • X'180A' - Получить информацию о первичном ключе
32	20	CHAR(10)	Библиотека таблицы с первичным ключом	Имя библиотеки, содержащей таблицу с первичным ключом, которая применяется для получения информации о первичном и внешнем ключе
42	2A	CHAR(128)	Таблица с первичным ключом (псевдоним)	Имя таблицы, содержащей первичный ключ, которая применяется для получения информации о первичном или внешнем ключе
170	AA	CHAR(10)	Библиотека таблицы с внешним ключом	Имя библиотеки, содержащей таблицу с внешним ключом, которая применяется для получения информации о внешнем ключе
180	64	CHAR(128)	Имя таблицы с внешним ключом (псевдоним)	Имя таблицы, содержащей внешний ключ, которая применяется для получения информации о внешнем ключе
<p>Примечание: Определение этого формата содержится в элементе EZDAEP файлов H, QRPGSRC, QRPGLSRC, QLBLSRC и QCBLLSRC, расположенных в библиотеке QSYSINC.</p>				

Сервер очередей данных: Для сервера очередей данных определена одна точка выхода:

QIBM_QZHQ_DATA_QUEUE в формате ZHQ00100

Точка выхода QIBM_QZHQ_DATA_QUEUE предназначена для вызова программы выхода, обрабатывающей следующие запросы к серверу очередей данных:

- Запросить
- Получить
- Создать
- Удалить
- Отправить
- Очистить
- Отменить
- Считать

Точка выхода QIBM_QZHQ_DATA_QUEUE, формат ZHQ00100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для сервера очередей данных это значение равно *DATAQSRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZHQ_DATA_QUEUE имя формата равно ZHQ00100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена <ul style="list-style-type: none"> • X'0001' - Получить атрибуты очереди данных • X'0002' - Получить сообщение из очереди данных • X'0003' - Создать очередь данных • X'0004' - Удалить очередь данных • X'0005' - Отправить сообщение в очередь данных • X'0006' - Очистить очередь данных • X'0007' - Отменить ожидающий запрос на получение информации • X'0012' - Получить сообщение из очереди данных, не удаляя его
32	20	CHAR(10)	Имя объекта	Имя очереди данных
42	2A	CHAR(10)	Имя библиотеки	Библиотека очереди данных
52	34	CHAR(2)	Оператор отношения	Оператор отношения для операции получения данных по ключу <ul style="list-style-type: none"> X'0000' - Нет оператора 'EQ' - Равно 'NE' - Не равно 'GE' - Больше или равно 'GT' - Больше 'LE' - Меньше или равно 'LT' - Меньше
54	36	BINARY(4)	Длина ключа	Длина ключа, заданная в запросе
58	3A	CHAR(256)	Значение ключа	Ключ, указанный в запросе
<p>Примечание: Определение этого формата содержится в элементе EZHQEP файлов H, QRPGSRC, QRPGLSRC, QLBSLRC и QCBLLSRC, расположенных в библиотеке QSYSINC.</p>				

Сервер сетевой печати: Для сервера сетевой печати определены две точки выхода:

1. QIBM_QNPS_ENTRY в формате ENTR0100
 - Применяется при инициализации сервера
2. QIBM_QNPS_SPLF в формате SPLF0100
 - Применяется для обработки буферного файла вывода

Точка выхода QIBM_QNPS_ENTRY предназначена для вызова программы выхода при запуске сервера сетевой печати. Эта программа выхода может применяться для проверки прав доступа к серверу. Дополнительная информация об этом приведена в книге *Printer Device Programming*, SC41-5713-03.

Точка выхода QIBM_QNPS_ENTRY, формат ENTR0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользов. профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для сервера сетевой печати это значение равно QNPSERVER.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QNPS_ENTRY применяется формат ENTR0100.
28	1C	BINARY(4)	Идентификатор функции	Функция, которая будет выполнена Для QIBM_QNPS_ENTRY это значение равно X'0802'.
<p>Примечание: Определение этого формата содержится в элементе ENPSEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLSRC, расположенных в библиотеке QSYSINC.</p>				

Точка выхода QIBM_QNPS_SPLF предназначена для запуска программы выхода при получении сервером сетевой печати запроса на обработку буферного файла вывода. Эта программа может применяться для выполнения определенных операций над буферным файлом, например, для его отправки по факсу. Дополнительная информация об этом приведена в книге *Printer Device Programming*, SC41-5713-03.

Точка выхода QIBM_QNPS_SPLF, формат SPLF0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для сервера сетевой печати это значение равно QNPSERVER
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QNPS_SPLF имя формата равно SPLF0100.
28	1C	BINARY(4)	Идентификатор функции	Функция, которая будет выполнена Для QIBM_QNPS_SPLF это значение равно X'010D'.
32	20	CHAR(10)	Имя задания	Имя задания, создавшего буферный файл
42	2A	CHAR(10)	Имя пользователя	Пользовательский профайл задания, создавшего буферный файл
52	34	CHAR(6)	Номер задания	Номер задания, создавшего буферный файл
58	3A	CHAR(10)	Имя буферного файла	Имя буферного файла, над которым нужно выполнить операцию

Смещение		Тип	Поле	Описание
Дес	Шест			
68	44	BINARY(4)	Номер буферного файла	Номер буферного файла, над которым нужно выполнить операцию
72	48	BINARY(4)	Длина	Длина данных программы выхода для буферного файла
76	4C	CHAR(*)	Данные программы выхода для буферного файла	Данные программы выхода для буферного файла представляют собой дополнительную информацию, которая применяется программой выхода, зарегистрированной для точки выхода QIBM_QNPS_SPLF. Эти данные предоставляются приложением клиента.
Примечание: Определение этого формата содержится в элементе ENPSEP файлов H, QRPGRSRC, QRPGLSRC, QLBLSRC и QCBLLESRC, расположенных в библиотеке QSYSINC.				

Центральный сервер: Для центрального сервера определены три точки выхода:

1. QIBM_QZSC_LM в формате ZSCL0100
 - Применяется для обработки запросов на управление лицензиями
2. QIBM_QZSC_SM в формате ZSCS0100
 - Применяется для обработки запросов на управление системой
3. QIBM_QZSC_NLS в формате ZSCN0100
 - Применяется для обработки запросов на получение таблиц преобразования

Точка выхода QIBM_QZSC_LM применяется для вызова программы выхода, обрабатывающей все запросы на управление лицензиями, отправляемые центральному серверу.

Программа выхода QIBM_QZSC_LM, формат ZSCL0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для центрального сервера это значение равно *CNTRLSRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZSC_LM имя формата равно ZSCL0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена Данное поле содержит одно из следующих значений: <ul style="list-style-type: none"> • X'1001' - Запрос на получение лицензии • X'1002' - Запрос на освобождение лицензии • X'1003' - Запрос на получение информации о лицензии

Смещение		Тип	Поле	Описание
Дес	Шест			
32	20	CHAR(255)	Уникальное имя клиента	Имя клиента применяется для идентификации рабочей станции в сети. Лицензия на продукт выдается рабочей станции с определенным именем клиента.
287	11F	CHAR(8)	Описатель владельца лицензии	С помощью описателя владельца лицензии система проверяет, что лицензию запросил и освободил один и тот же пользователь. Значение, указанное при освобождении лицензии, должно совпадать со значением, заданным при получении лицензии.
295	127	CHAR(7)	Идентификатор продукта	Идентификатор продукта, лицензия на который была запрошена
302	12E	CHAR(4)	Идентификатор компонента	Компонент продукта
306	132	CHAR(6)	Выпуск	Версия, выпуск и уровень модификации продукта или его компонента
312	138	BINARY(2)	Тип информации	<p>Тип возвращаемой информации.</p> <p>Тип информации указывается только в том случае, если был получен запрос на получение информации о лицензии</p> <p>Данное поле содержит одно из следующих значений:</p> <ul style="list-style-type: none"> • X'0000' - Основная информация о лицензии • X'0001' - Подробная информация о лицензии
<p>Примечание: Определение этого формата содержится в элементе EZSCEP файлов H, QRPGSRC, QRPGLSRC, QLBSLRC и QCBLLSRC, расположенных в библиотеке QSYSINC.</p>				

Точка выхода QIBM_QZSC_SM предназначена для вызова программы выхода, обрабатывающей все запросы на управление клиентами, отправляемые центральному серверу.

Программа выхода QIBM_QZSC_SM, формат ZSCS0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для центрального сервера это значение равно *CNTRLSRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZSC_SM имя формата равно ZSCS0100.

Смещение		Тип	Поле	Описание
Дес	Шест			
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена Данное поле содержит одно из следующих значений: • X'1101' - Активировать клиент • X'1102' - Деактивировать клиент
32	20	CHAR(255)	Уникальное имя клиента	Имя рабочей станции клиента, связанное с лицензионным продуктом
287	11F	CHAR(255)	Имя группы взаимодействия	Этот параметр конфигурации SNMP применяется для идентификации.
542	21E	CHAR(1)	Тип узла	Тип соединения • 3 - Internet
543	21F	CHAR(255)	Имя узла	Имя узла Для узлов типа 3 в качестве имени указывается их IP-адрес.
<p>Примечание: Определение этого формата содержится в элементе EZSCEP файлов H, QRPGRSRC, QRPGLSRC, QLBSLRC и QCBLLESRC, расположенных в библиотеке QSYSINC.</p>				

Точка выхода QIBM_QZSC-NLS предназначена для вызова программы выхода, обрабатывающей запросы на получение таблицы преобразования, отправляемые центральному серверу.

Программа выхода QIBM_QZSC-NLS, формат ZSCN0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для центрального сервера это значение равно *CNTRLRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZSC-NLS имя формата равно ZSCN0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена • X'1201' - Получить таблицу преобразования
32	20	BINARY(4)	Идентификатор исходного набора символов (CCSID)	CCSID исходных данных
36	24	BINARY(4)	Идентификатор целевого набора символов (CCSID)	CCSID, в который должны быть преобразованы данные
40	28	BINARY(2)	Тип преобразования	Запрошенный тип преобразования: • X'0001' - Циклическое • X'0002' - Путем подстановки • X'0003' - Путем поиска оптимального варианта

Смещение		Тип	Поле	Описание
Дес	Шест			
Примечание: Определение этого формата содержится в элементе EZSCEP файлов H, QRPGSRC, QRPGLSRC, QLBSLRC и QCBLESRC, расположенных в библиотеке QSYSINC.				

Сервер обработки удаленных команд и вызовов распределенных программ: Для сервера обработки удаленных команд и вызовов распределенных программ определена одна точка выхода:

QIBM_QZRC_RMT в формате CZRC0100

Точка выхода QIBM_QZRC_RMT предназначена для запуска программы при получении запроса на вызов удаленной команды или распределенной программы.

Формат параметров программы выхода зависит от типа запроса.

Запросы на запуск удаленной команды, точка выхода QIBM_QZRC_RMT, формат CZRC0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для сервера обработки удаленных команд это значение равно *RMTSRV.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZRC_RMT имя формата равно CZRC0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена X'1002' - Удаленная команда
32	20	CHAR(10)	Зарезервировано	Не применяется для запросов на запуск удаленных команд
42	2A	CHAR(10)	Зарезервировано	Не применяется для запросов на запуск удаленных команд
52	34	BINARY(4)	Длина следующего поля	Длина команды, указанной в следующем поле
56	38	CHAR (6000)	Команда	Командная строка для вызова удаленной команды

Запросы на вызов распределенной программы, точка выхода QIBM_QZRC_RMT, формат CZRC0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, отправившего запрос серверу
10	A	CHAR(10)	Идентификатор сервера	Для сервера обработки вызовов распределенных программ это значение равно *RMTSRV.

Смещение		Тип	Поле	Описание
Дес	Шест			
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZRC_RMT имя формата равно CZRC0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена X'1003' - Вызов распределенной программы
32	20	CHAR(10)	Имя программы	Имя программы, которая будет вызвана
42	2A	CHAR(10)	Имя библиотеки	Библиотека, в которой расположена указанная программа
52	34	BINARY(4)	Число параметров	Общее число параметров программы. Оно не всегда совпадает с фактическим числом параметров.
56	38	CHAR(*)	Информация о параметрах	Информация о параметрах, передаваемых указанной программе. Все записи о параметрах задаются в одном и том же формате, независимо от их типа. Тип параметра (входной или выходной) указывается в последнем поле структуры. <ul style="list-style-type: none"> • BINARY(4) - Длина информации о данном параметре • BINARY(4) - Максимальный размер параметра • BINARY(2) - Тип параметра <ul style="list-style-type: none"> — 1 - Входной — 2 - Выходной — 3 - Входной/выходной • CHAR(*) - Строка с информацией о параметре <p>Максимальный размер информации о параметре составляет 6000 байт. Все лишние данные отсекаются.</p>

Сервер входа в систему: Для сервера входа в систему определена одна точка выхода:

QIBM_QZSO_SIGNONSRV в формате ZSOY0100

Точка выхода QIBM_QZSO_SIGNONSRV предназначена для вызова программы выхода при получении сервером входа в систему следующих запросов:

- Получить информацию о входе в систему
- Изменить пароль
- Создать разрешение

Точка выхода QIBM_QZSO_SIGNONSRV, формат ZSOY0100

Смещение		Тип	Поле	Описание
Дес	Шест			
0	0	CHAR(10)	Имя пользовательского профайла	Имя пользовательского профайла, связанного с запросом

Смещение		Тип	Поле	Описание
Дес	Шест			
10	A	CHAR(10)	Идентификатор сервера	Для сервера входа в систему это значение равно *SIGNON.
20	14	CHAR(8)	Имя формата	Имя формата, связанное с пользовательской программой выхода. Для QIBM_QZSO_SIGNONSRV имя формата равно ZSOY0100.
28	1C	BINARY(4)	Запрошенная функция	Функция, которая будет выполнена <ul style="list-style-type: none"> • X'7004' - Получить информацию о входе в систему • X'7005' - Изменить пароль • X'7007' - Создать разрешение

Примеры: Программы выхода

В приведенных примерах программ выхода продемонстрированы далеко не все приемы программирования, однако вам будет полезно ознакомиться с ними перед созданием собственных программ.

Отказ от гарантий на предоставляемый код

Фирма IBM предоставляет вам неисключительное право на использование всех этих примеров программного кода, на основе которых вы можете создавать собственные программы.

Все указанные примеры кода приведены фирмой IBM исключительно для иллюстрации. Они не были тщательно и всесторонне протестированы. По этой причине фирма IBM не может гарантировать их надежность, удобство их обслуживания и отсутствие в них ошибок.

Все приведенные программы предоставляются на условиях "КАК ЕСТЬ" без каких-либо гарантий, включая гарантии соблюдения прав, коммерческой ценности и пригодности для конкретных целей.

- Примеры: Создание программ выхода на языке RPG
- Примеры: Создание программ выхода на языке CL

Примеры: Создание программ выхода на: языке RPG

Ниже приведен пример пользовательской программы выхода на языке RPG*.

Примечание: ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

**

** Сервер OS/400 - Пример пользовательской программы выхода

**

** Приведенная ниже программа на языке RPG принимает

** все запросы. Ее можно использовать в качестве основы

** для создания собственных приложений. Примечание: Для

** сокращения объема кода удалите функции и операторы CASE

** для тех серверов, запросы к которым не должны обрабатываться

** программой выхода.

**

E*

E* Определения массивов для функции передачи данных и удаленных запросов

E* SQL

E*

```

E          TFREQ    4096  1
E          RSREQ    4107  1
I*
I*
IPCSDTA      DS
I          1  10  USERID
I          11 20  APPLID
I*
I* Параметры для виртуального принтера
I*
I          21 30  VPFUNC
I          31 40  VPOBJ
I          41 50  VPLIB
I          71 750VPIFN
I          76 85  VPOUTQ
I          86 95  VPQLIB
I*
I* Параметры для функции отправки сообщений
I          21 30  MFFUNC
I*
I* Параметры для функции передачи данных
I*
I          21 30  TFFUNC
I          31 40  TFOBJ
I          41 50  TFLIB
I          51 60  TFMBR
I          61 70  TFFMT
I          71 750TFLEN
I          764171 TFREQ
I*
I* Параметры для файлового сервера
I*
I* Примечание: Размер FSNAME не должен превосходить 16 Мб.
I* Фактический размер FSNAME хранится в переменной FSNLEN.
I*
I          B 21 240FSFID
I          25 32  FSFMT
I          33 33  FSREAD
I          34 34  FSWRIT
I          35 35  FSRDWR
I          36 36  FSDLT
I          B 37 400FSNLEN
I          41 296 FSNAME
I*
I* Параметры для очередей данных
I*
I          21 30  DQFUNC
I          31 40  DQQ
I          41 50  DQLIB
I          70 750DQLEN
I          76 77  DQROP
I          78 820DQKLEN
I          83 338 DQKEY
I*
I* Параметры для удаленных запросов SQL
I*
I          21 30  RSFUNC
I          31 40  RSOBJ
I          41 50  RSLIB
I          51 51  RSCMT
I          52 52  RSMODE
I          53 53  RSCID
I          54 71  RSSTN
I          72 75  RSRSV
I          764182 RSREQ
I*
I* Параметры для сервера сетевой печати

```

```

I*
I          21 28 NPFT
I          B 29 320NPFFID
I* Перечисленные ниже параметры относятся к формату SPLF0100
I          33 42 NPJOBN
I          43 52 NPUSRN
I          53 58 NPJOB#
I          59 68 NPFILE
I          B 69 720NPFIL#
I          B 73 760NPLEN
I          77 332 NPDATA
I*
I* Сервер очередей данных:
I*
I* QIBM_QZHQ_DATA_QUEUE, формат ZHQ00100
I*
I          21 28 DQOFMT
I          B 29 320DQOFID
I          33 42 DQO0BJ
I          43 52 DQOLIB
I          53 54 DQOROP
I          B 55 580DQOLEN
I          59 314 DQOKEY
I*
I* Параметры для центрального сервера
I*
I          21 28 CSFMT
I          B 29 320CSFID
I* Центральный сервер:
I*
I* QIBM_QZSC_LM, формат ZSCL0100 (для запросов к функции управления лицензиями)
I*
I*
I          33 287 CSLCNM
I          288 295 CSLUSR
I          296 302 CSLPID
I          303 306 CSLFID
I          307 312 CSLRID
I          B 313 3140CSLTYP
I*
I* Центральный сервер:
I*
I* QIBM_QZSC_LM, формат ZSCS0100 (для запросов к функции управления системой)
I*
I*
I          33 287 CSSCNM
I          288 542 CSSCMY
I          543 543 CSSNDE
I          544 798 CSSNNM
I*
I* Центральный сервер:
I*
I* QIBM_QZSC_LM, формат ZSCN0100 (для запросов на получение таблицы преобразования)
I*
I*
I          21 30 CSNXFM
I          29 320CSNFNC
I          B 33 360CSNFRM
I          B 37 400CSNTO
I          B 41 420CSNCNT
I*
I* Параметры для сервера баз данных
I*
I          21 28 DBFMT
I          B 29 320DBFID
I*

```

```

I* Перечисленные ниже параметры относятся к формату ZDAD0100
I          33 160 DBDFIL
I          161 170 DBDLIB
I          171 180 DBDMBR
I          181 190 DBDAUT
I          191 318 DBDBFL
I          319 328 DBDBLB
I          329 338 DBDOFL
I          339 348 DBDOLB
I          349 358 DBDOMB
I*
I* Перечисленные ниже параметры относятся к формату ZDAD0200
I          B 33 360DBNUM
I          37 46 DBLIB2
I*
I* Перечисленные ниже параметры относятся к формату ZDAQ0100
I          33 50 DBSTMT
I          51 68 DBCRSR
I          69 70 DBOPI
I          71 72 DBATTR
I          73 82 DBPKG
I          83 92 DBPLIB
I          B 93 940DBDRDA
I          95 95 DBCMT
I          96 351 DBTEXT
I* Перечисленные ниже параметры заменяют DBTEXT для формата ZDAQ0200
I          96 105 DBSQCL
I          B 133 1360DBSQLN
I          137 392 DBSQTX
I* Перечисленные ниже параметры относятся к формату ZDAR0100
I          33 52 DBLIBR
I          53 88 DBRDBN
I          89 108 DBPKGR
I          109 364 DBFILR
I          365 384 DBMBRR
I          385 404 DBFFT
I* Перечисленные ниже параметры относятся к формату ZDAR0200
I          33 42 DBRPLB
I          43 170 DBRPTB
I          171 180 DBRFLB
I          181 308 DBRFTB
I*
I* Сервер обработки удаленных команд и вызовов распределенных программ:
I*
I* QIBM_QZRC_RMT, формат CZRC0100
I*   Параметры RCPGM и RCLIB не применяются для обработки запросов на запуск удаленных команд
I*
I          21 28 RCFMT
I          B 29 320RCFID
I          33 42 RCPGM
I          43 52 RCLIB
I          B 53 560RCNUM
I          57 312 RCDATA
I*
I* Сервер входа в систему:
I*
I* QIBM_QZSO_sign-onSRV, формат ZSOY0100 (для сервера входа в систему TCP/IP)
I*
I          21 28 SOXFMT
I          B 29 320SOFID
I*
I*****
I*
I          '*VPRT      '      C          #VPRT
I          '*TRFCL   '      C          #TRFCL
I          '*FILESRV '      C          #FILE
I          '*MSGFCL  '      C          #MSGF

```

```

I          '*DQSRV '          C          #DQSRV
I          '*RQSRV '          C          #RQSRV
I          '*SQL '           C          #SQL
I          '*NDB '           C          #NDBSV
I          '*SQLSRV '        C          #SQLSV
I          '*RTVOBJINF'      C          #RTVOB
I          '*DATAQSRV '      C          #DATAQ
I          '*QNPSERV '       C          #QNPSV
I          '*CNTRLSRV '      C          #CNTRL
I          '*RMTSRV '       C          #RMTSV
I          '*sign-on '       C          #SIGN
I*
C*
C* Параметры программы выхода
C*
C          *ENTRY    PLIST
C                      PARM          RTNCD  1
C                      PARM          PCSDTA
C*
C* Устанавливается код возврата, разрешающий прием запроса
C*
C                      MOVE '1'      RTNCD
C*
C* Общий фрагмент кода
C*
C                      Произвольный код
C*
C* Определение ИД сервера
C*
C          APPLID    CASEQ#VPRT      VPRT
C          APPLID    CASEQ#TRFCL     TFR
C          APPLID    CASEQ#FILE      FILE
C          APPLID    CASEQ#MSGF      MSG
C          APPLID    CASEQ#DQSRV     DATAQ
C          APPLID    CASEQ#RQSRV     RSQL
C          APPLID    CASEQ#SQL       SQLINT
C          APPLID    CASEQ#NDBSV     NDB
C          APPLID    CASEQ#SQLSV     SQLSRV
C          APPLID    CASEQ#RTVOB     RTVOBJ
C          APPLID    CASEQ#DATAQ     ODATAQ
C          APPLID    CASEQ#QNPSV     NETPRT
C          APPLID    CASEQ#CNTRL     CENTRL
C          APPLID    CASEQ#RMTSV     RMTCMD
C          APPLID    CASEQ#SIGN      sign-on
C          END
C          SETON                      LR
C          RETRN
C*
C* Функции
C*
C*
C* Виртуальный принтер
C*
C          VPRT      BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Функция передачи данных
C*
C* Ниже приведен пример действий, которые программа выхода
C* может выполнять для функции передачи данных.
C*
C* В данном случае пользователям запрещается считывать данные из файлов,
C* расположенных в библиотеке QIWS.
C*
C          TFR      BEGSR
C          TFFUNC   IFEQ 'SELECT'

```

```

C          TFLIB      ANDEQ'QIWS'
C          MOVE '0'      RTNCD
C          END
C          ENDSR
C*
C*
C* Файловый сервер
C*
C          FILE      BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Функция отправки сообщений
C*
C          MSG      BEGSR
C*          Фрагмент кода
C          ENDSR
C* Очереди данных
C*
C          DATAQ    BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Удаленные запросы SQL
C*
C          RSQL      BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Серверы
C*
C* Инициализация базы данных
C*
C          SQLINT    BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Внутренние запросы к базе данных
C*
C          NDB      BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Запросы SQL к базе данных
C*
C          SQLSRV    BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Получение информации об объекте
C*
C          RTVOBJ    BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Сервер очередей данных
C*
C          ODATAQ    BEGSR
C*          Фрагмент кода
C          ENDSR
C*
C* Сервер сетевой печати
C*
C          NETPRT    BEGSR
C*          Фрагмент кода
C          ENDSR

```

```

C*
C* Центральный сервер
C*
C*
C* Ниже приведен пример действий, которые программа выхода
C* может выполнять для запросов к функции управления лицензиями.
C*
C* В данном случае пользователю "USERALL" запрещается
C* доступ ко всем функциям центрального сервера, с которым связана
C* эта программа выхода, в том числе к информации о лицензиях,
C* функциям управления системой и функции отправки таблицы
C* преобразования.
C*
C          CENTRL  BEGSR
C          USERID  IFEQ 'USERALL'
C                   MOVE '0'          RTNCD
C                   ENDIF
C*          Фрагмент кода
C          ENDSR
C*
C* Сервер обработки удаленных команд и вызовов распределенных программ
C*
C* В данном случае пользователю "USERALL" запрещается
C* запускать удаленные команды и вызывать программы
C*
C          RMTCMD  BEGSR
C          USERID  IFEQ 'USERALL'
C                   MOVE '0'          RTNCD
C                   ENDIF
C          ENDSR
C*
C* sign-on SERVER
C*
C          sign-on  BEGSR
C*          Фрагмент кода
C          ENDSR

```

Примеры: Создание программ выхода на языке CL: Ниже приведен пример пользовательской программы выхода на языке CL.

Примечание: ознакомьтесь с важной юридической информацией, приведенной в разделе Отказ от гарантий на предоставляемый код.

```

/*****
/*
/* Сервер iSeries - Пример пользовательской программы выхода      */
/*
/* Данная программа на языке CL принимает все запросы без        */
/* исключения. Ее можно использовать в качестве основы для      */
/* создания программ выхода для своей системы.                   */
/*
/*
/*
/*****
PGM PARM(&STATUS &REQUEST)

/* * * * * * * * * * * * * * * * * * * * * * * * * * * */
/*
/* Объявление параметров программы                               */
/*
/* * * * * * * * * * * * * * * * * * * * * * * * * * * */

DCL VAR(&STATUS) TYPE(*CHAR) LEN(1) /* Индикатор Запрос принят/Запрос отклонен */
/* */
/* Примечание: Переменная REQUEST объявлена как *CHAR LEN(2000), так как это максимальная */

```

```

/* длина строки в языке CL. Фактический размер переменной REQUEST равен 4171. */
/* */
DCL VAR(&REQUEST) TYPE(*CHAR) LEN(2000) /* Структура данных */
/*****/
/* */
/* Объявление параметров */
/* */
/*****/

/* Объявление общих параметров */
DCL VAR(&USER) TYPE(*CHAR) LEN(10)
/* ИД пользователя */
DCL VAR(&APPLIC) TYPE(*CHAR) LEN(10)
/* ИД сервера */
DCL VAR(&FUNCTN) TYPE(*CHAR) LEN(10) /* Запрошенная функция */

/* Объявление параметров виртуального принтера */
DCL VAR(&VPOBJ) TYPE(*CHAR) LEN(10) /* Имя объекта */
DCL VAR(&VPLIB) TYPE(*CHAR) LEN(10) /* Библиотека объекта */
DCL VAR(&VPLEN) TYPE(*DEC) LEN(5 0) /* Размер следующих полей */
DCL VAR(&VPOUTQ) TYPE(*CHAR) LEN(10) /* Имя очереди вывода */
DCL VAR(&VPLIB) TYPE(*CHAR) LEN(10) /* Библиотека очереди вывода */

/* Объявление переменных функции передачи данных */
DCL VAR(&TFOBJ) TYPE(*CHAR) LEN(10) /* Имя объекта */
DCL VAR(&TFLIB) TYPE(*CHAR) LEN(10) /* Библиотека объекта */
DCL VAR(&TFMBR) TYPE(*CHAR) LEN(10) /* Имя элемента */
DCL VAR(&TFMT) TYPE(*CHAR) LEN(10) /* Имя формата записи */
DCL VAR(&TFLEN) TYPE(*DEC) LEN(5 0) /* Длина запроса */
DCL VAR(&TFREQ) TYPE(*CHAR) LEN(1925) /* Оператор для передачи запроса*/

/* Объявление переменных файлового сервера */
DCL VAR(&FSFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */
DCL VAR(&FSFMT) TYPE(*CHAR) LEN(8) /* Формат параметра */
DCL VAR(&FSREAD) TYPE(*CHAR) LEN(1) /* Открыть для чтения */
DCL VAR(&FSWRITE) TYPE(*CHAR) LEN(1) /* Открыть для записи */
DCL VAR(&FSRDWRT) TYPE(*CHAR) LEN(1) /* Открыть для чтения и записи */
DCL VAR(&FSDLT) TYPE(*CHAR) LEN(1) /* Открыть для удаления */
DCL VAR(&FSLEN) TYPE(*CHAR) LEN(4) /* Длина переменной fname */
DCL VAR(&FSNAME) TYPE(*CHAR) LEN(2000) /* Полное имя файла */

/* Объявление переменных для очереди данных */
DCL VAR(&DQQ) TYPE(*CHAR) LEN(10) /* Имя очереди данных */
DCL VAR(&DQLIB) TYPE(*CHAR) LEN(10) /* Библиотека очереди данных */
DCL VAR(&DQLEN) TYPE(*DEC) LEN(5 0) /* Общая длина запроса */
DCL VAR(&DQRDP) TYPE(*CHAR) LEN(2) /* Оператор отношения */
DCL VAR(&DQKLEN) TYPE(*DEC) LEN(5 0) /* Длина ключа */
DCL VAR(&DQKEY) TYPE(*CHAR) LEN(256) /* Ключ */

/* Объявление переменных для удаленных запросов SQL */
DCL VAR(&RSOBJ) TYPE(*CHAR) LEN(10) /* Имя объекта */
DCL VAR(&RSLIB) TYPE(*CHAR) LEN(10) /* Библиотека объекта */
DCL VAR(&RSCMT) TYPE(*CHAR) LEN(1) /* Уровень управления фиксацией */
DCL VAR(&RSMODE) TYPE(*CHAR) LEN(1) /* Индикатор режима работы с блоками или обновления */
DCL VAR(&RSCID) TYPE(*CHAR) LEN(1) /* ИД курсора */
DCL VAR(&RSSTN) TYPE(*CHAR) LEN(18) /* Имя оператора */
DCL VAR(&RSRSU) TYPE(*CHAR) LEN(4) /* Зарезервирован */
DCL VAR(&RSREQ) TYPE(*CHAR) LEN(1925) /* Оператор SQL */

/* Объявление переменных для сервера сетевой печати */
DCL VAR(&NPFMT) TYPE(*CHAR) LEN(8) /* Имя формата */
DCL VAR(&NPFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */
/* Все остальные параметры относятся только к формату SPLF0100 */
DCL VAR(&NPJOBN) TYPE(*CHAR) LEN(10) /* Имя задания */
DCL VAR(&NPUSRN) TYPE(*CHAR) LEN(10) /* Имя пользователя */
DCL VAR(&NPJOB#) TYPE(*CHAR) LEN(6) /* Номер задания */
DCL VAR(&NPFILE) TYPE(*CHAR) LEN(10) /* Имя файла */

```

```

DCL VAR(&NPFIL#) TYPE(*CHAR) LEN(4) /* Номер файла */
DCL VAR(&NPLEN) TYPE(*CHAR) LEN(4) /* Размер данных */
DCL VAR(&NPDATA) TYPE(*CHAR) LEN(2000) /* Данные */

DCL VAR(&DBNUM) TYPE(*CHAR) LEN(4) /* Число библиотек */
DCL VAR(&DBLIB2) TYPE(*CHAR) LEN(10) /* Имя библиотеки */

/* Объявление переменных для сервера очередей данных */
DCL VAR(&DQFMT) TYPE(*CHAR) LEN(8) /* Имя формата */
DCL VAR(&DQFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */
DCL VAR(&DQOBJ) TYPE(*CHAR) LEN(10) /* Имя объекта */
DCL VAR(&DQOLIB) TYPE(*CHAR) LEN(10) /* Имя библиотеки */
DCL VAR(&DQOROP) TYPE(*CHAR) LEN(2) /* Оператор отношения */
DCL VAR(&DQOLEN) TYPE(*CHAR) LEN(4) /* Длина ключа */
DCL VAR(&DQOKEY) TYPE(*CHAR) LEN(256) /* Ключ */

/* Объявление переменных для центрального сервера */
DCL VAR(&CSFMT) TYPE(*CHAR) LEN(8) /* Имя формата */
DCL VAR(&CSFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */
/* Перечисленные ниже параметры относятся к формату ZSCL0100 */
DCL VAR(&CSCNAM) TYPE(*CHAR) LEN(255) /* Уникальное имя клиента */
DCL VAR(&CSLUSR) TYPE(*CHAR) LEN(8) /* Описатель пользователя лицензии */
DCL VAR(&CSPID) TYPE(*CHAR) LEN(7) /* Идентификатор продукта */
DCL VAR(&CSFID) TYPE(*CHAR) LEN(4) /* Идентификатор компонента */
DCL VAR(&CSRID) TYPE(*CHAR) LEN(6) /* Выпуск */
DCL VAR(&CSTYPE) TYPE(*CHAR) LEN(2) /* Тип запрошенной информации */
/* Перечисленные ниже параметры относятся к формату ZSCS0100 */
DCL VAR(&CSCNAM) TYPE(*CHAR) LEN(255) /* Уникальное имя клиента */
DCL VAR(&CSCMTY) TYPE(*CHAR) LEN(255) /* Имя группы взаимодействия */
DCL VAR(&CSNODE) TYPE(*CHAR) LEN(1) /* Тип узла */
DCL VAR(&CSNNAM) TYPE(*CHAR) LEN(255) /* Имя узла */
/* Перечисленные ниже параметры относятся к формату ZSCN0100 */
DCL VAR(&CSFROM) TYPE(*CHAR) LEN(4) /* Исходный CCSID */
DCL VAR(&CSTO) TYPE(*CHAR) LEN(4) /* Целевой CCSID */
DCL VAR(&CSCTYP) TYPE(*CHAR) LEN(2) /* Тип преобразования */
/* Объявление переменных для сервера баз данных */
DCL VAR(&DBFMT) TYPE(*CHAR) LEN(8) /* Имя формата */
DCL VAR(&DBFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */

/* Перечисленные ниже параметры относятся к формату ZDAD0100 */
DCL VAR(&DBFILE) TYPE(*CHAR) LEN(128) /* Имя файла */
DCL VAR(&DBLIB) TYPE(*CHAR) LEN(10) /* Имя библиотеки */
DCL VAR(&DBMBR) TYPE(*CHAR) LEN(10) /* Имя элемента */
DCL VAR(&DBAUT) TYPE(*CHAR) LEN(10) /* Права доступа к файлу */
DCL VAR(&DBBFIL) TYPE(*CHAR) LEN(128) /* Имя исходного файла */
DCL VAR(&DBBLIB) TYPE(*CHAR) LEN(10) /* Библиотека исходного файла */
DCL VAR(&DBOFIL) TYPE(*CHAR) LEN(10) /* Имя переопределяемого файла */
DCL VAR(&DBOLIB) TYPE(*CHAR) LEN(10) /* Библиотека переопределяемого файла */
DCL VAR(&DBOMBR) TYPE(*CHAR) LEN(10) /* Имя переопределяемого элемента */

/* Перечисленные ниже параметры относятся к формату ZDAD0200 */
DCL VAR(&DBNUM) TYPE(*CHAR) LEN(4) /* Число библиотек */
DCL VAR(&DBLIB2) TYPE(*CHAR) LEN(10) /* Имя библиотеки */

/* Перечисленные ниже параметры относятся к формату ZDAQ0100 */
DCL VAR(&DBSTMT) TYPE(*CHAR) LEN(18) /* Имя оператора */
DCL VAR(&DBCRSR) TYPE(*CHAR) LEN(18) /* Имя курсора */
DCL VAR(&DBOPT) TYPE(*CHAR) LEN(2) /* Опция подготовки */
DCL VAR(&DBATTR) TYPE(*CHAR) LEN(2) /* Атрибуты функции открытия */
DCL VAR(&DBPKG) TYPE(*CHAR) LEN(10) /* Имя пакета */
DCL VAR(&DBPLIB) TYPE(*CHAR) LEN(10) /* Библиотека пакета */
DCL VAR(&DBDRDA) TYPE(*CHAR) LEN(2) /* Индикатор DRDA */
DCL VAR(&DBCMT) TYPE(*CHAR) LEN(1) /* Уровень управления фиксацией */
DCL VAR(&DBTEXT) TYPE(*CHAR) LEN(512) /* Первые 512 байт текста оператора */

```

```

/* Перечисленные ниже параметры относятся к формату ZDAR0100 */
DCL VAR(&DBLIBR) TYPE(*CHAR) LEN(20) /* Имя библиотеки */
DCL VAR(&DBRDBN) TYPE(*CHAR) LEN(36) /* Имя реляционной базы данных */
DCL VAR(&DBPKGR) TYPE(*CHAR) LEN(20) /* Имя пакета */
DCL VAR(&DBFILR) TYPE(*CHAR) LEN(256) /* Имя файла (псевдоним SQL) */
DCL VAR(&DBMBRR) TYPE(*CHAR) LEN(20) /* Имя элемента */
DCL VAR(&DBFFMT) TYPE(*CHAR) LEN(20) /* Имя формата */

/* Перечисленные ниже параметры относятся к формату ZDAR0200 */
DCL VAR(&DBPLIB) TYPE(*CHAR) LEN(10) /* Библиотека таблицы с первичным ключом */
DCL VAR(&DBPTBL) TYPE(*CHAR) LEN(128) /* Таблица с первичным ключом */
DCL VAR(&DBFLIB) TYPE(*CHAR) LEN(10) /* Библиотека таблицы с внешним ключом */
DCL VAR(&DBFTBL) TYPE(*CHAR) LEN(128) /* Таблица с внешним ключом */

/* Объявление переменных для сервера обработки удаленных команд */
DCL VAR(&RCFMT) TYPE(*CHAR) LEN(8) /* Имя формата */
DCL VAR(&RCFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */
DCL VAR(&RCPGM) TYPE(*CHAR) LEN(10) /* Имя программы */
DCL VAR(&RCLIB) TYPE(*CHAR) LEN(10) /* Библиотека программы */
DCL VAR(&RCNUM) TYPE(*CHAR) LEN(4) /* Число параметров или длина команды */
DCL VAR(&RCDATA) TYPE(*CHAR) LEN(6000) /* Команда или параметры */

/* Объявление переменных для сервера входа в систему */

DCL VAR(&SOFMT) TYPE(*CHAR) LEN(8) /* Имя формата */
DCL VAR(&SOFID) TYPE(*CHAR) LEN(4) /* Идентификатор функции */

/*****/
/* */
/* Объявление прочих переменных */
/* */
/*****/
DCL VAR(&WRKLEN) TYPE(*CHAR) LEN(5)
DCL VAR(&DECLEN) TYPE(*DEC) LEN(8 0)

/* * * * * * */
/* */
/* Получение параметров из структуры */
/* */
/* * * * * * */

/* Заголовок */
CHGVAR VAR(&USER) VALUE(%SST(&REQUEST 1 10))
CHGVAR VAR(&APPLIC) VALUE(%SST(&REQUEST 11 10))
CHGVAR VAR(&FUNCTN) VALUE(%SST(&REQUEST 21 10))

/* Виртуальный принтер */
CHGVAR VAR(&VPOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&VPLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&VPLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&VPOUTQ) VALUE(%SST(&REQUEST 76 10))
CHGVAR VAR(&VPQLIB) VALUE(%SST(&REQUEST 86 10))

/* Функция передачи данных */
CHGVAR VAR(&TFOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&TFLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&TFMBR) VALUE(%SST(&REQUEST 51 10))
CHGVAR VAR(&TFFMT) VALUE(%SST(&REQUEST 61 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&TFLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&TFREQ) VALUE(%SST(&REQUEST 76 1925))

/* Файловый сервер */
CHGVAR VAR(&FSFID) VALUE(%SST(&REQUEST 21 4))
CHGVAR VAR(&FSFMT) VALUE(%SST(&REQUEST 25 8))
CHGVAR VAR(&FSREAD) VALUE(%SST(&REQUEST 33 1))

```

```

CHGVAR VAR(&FSWRITE) VALUE(%SST(&REQUEST 34 1))
CHGVAR VAR(&FSRDWRT) VALUE(%SST(&REQUEST 35 1))
CHGVAR VAR(&FSDLT) VALUE(%SST(&REQUEST 36 1))
CHGVAR VAR(&FSLEN) VALUE(%SST(&REQUEST 37 4))
CHGVAR VAR(&DECLEN) VALUE(%BINARY(&FSLEN 1 4))
CHGVAR VAR(&FSNAME) VALUE(%SST(&REQUEST 41 &DECLEN))

/* Очереди данных */
CHGVAR VAR(&DQQ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&DQLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&DQLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&DQROP) VALUE(%SST(&REQUEST 76 2))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 78 5))
CHGVAR VAR(&DQKLEN) VALUE(&WRKLEN)
CHGVAR VAR(&DQKEY) VALUE(%SST(&REQUEST 83 &DQKLEN))

/* Удаленные запросы SQL */
CHGVAR VAR(&RSOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&RSLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&RSCMT) VALUE(%SST(&REQUEST 51 1))
CHGVAR VAR(&RSMODE) VALUE(%SST(&REQUEST 52 1))
CHGVAR VAR(&RSCID) VALUE(%SST(&REQUEST 53 1))
CHGVAR VAR(&RSSSTN) VALUE(%SST(&REQUEST 54 18))
CHGVAR VAR(&RSRSU) VALUE(%SST(&REQUEST 72 4))
CHGVAR VAR(&RSREQ) VALUE(%SST(&REQUEST 76 1925))

/* Сервер сетевой печати */
CHGVAR VAR(&NPFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&NPFID) VALUE(%SST(&REQUEST 29 4))

/* Если формат равен SPLF0100 */
IF COND(&NPFMT *EQ 'SPLF0100') THEN(DO)
CHGVAR VAR(&NPJOBN) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&NPUSRN) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&NPJOB#) VALUE(%SST(&REQUEST 53 6))
CHGVAR VAR(&NPFILE) VALUE(%SST(&REQUEST 59 10))
CHGVAR VAR(&NPFIL#) VALUE(%SST(&REQUEST 69 4))
CHGVAR VAR(&NPLEN) VALUE(%SST(&REQUEST 73 4))
CHGVAR VAR(&DECLEN) VALUE(%BINARY(&NPLEN 1 4))
CHGVAR VAR(&NPDATA) VALUE(%SST(&REQUEST 77 &DECLEN))
ENDDO

/* Сервер очередей данных */
CHGVAR VAR(&DQFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&DQFID) VALUE(%SST(&REQUEST 29 4))
CHGVAR VAR(&DQOOBJ) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&DQOLIB) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&DQOROP) VALUE(%SST(&REQUEST 53 2))
CHGVAR VAR(&DQOLEN) VALUE(%SST(&REQUEST 55 4))
CHGVAR VAR(&DQOKEY) VALUE(%SST(&REQUEST 59 256))

/* Центральный сервер */
CHGVAR VAR(&CSFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&CSFID) VALUE(%SST(&REQUEST 29 4))

/* Если формат равен ZSCL0100 */
IF COND(&CSFMT *EQ 'ZSCL0100') THEN(DO)
CHGVAR VAR(&CSCNAM) VALUE(%SST(&REQUEST 33 255))
CHGVAR VAR(&CSLUSR) VALUE(%SST(&REQUEST 288 8))
CHGVAR VAR(&CSPID) VALUE(%SST(&REQUEST 296 7))
CHGVAR VAR(&CSFID) VALUE(%SST(&REQUEST 303 4))
CHGVAR VAR(&CSRID) VALUE(%SST(&REQUEST 307 6))
CHGVAR VAR(&CSTYPE) VALUE(%SST(&REQUEST 313 2))
ENDDO

```

```

/* Если формат равен ZSCS0100 */
IF COND(&CSFMT *EQ 'ZSCS0100') THEN(DO)
  CHGVAR VAR(&CSCNAM) VALUE(%SST(&REQUEST 33 255))
  CHGVAR VAR(&CSCMTY) VALUE(%SST(&REQUEST 288 255))
  CHGVAR VAR(&CSNODE) VALUE(%SST(&REQUEST 543 1))
  CHGVAR VAR(&CSNNAM) VALUE(%SST(&REQUEST 544 255))
ENDDO

/* Если формат равен ZSCN0100 */
IF COND(&CSFMT *EQ 'ZSCN0100') THEN(DO)
  CHGVAR VAR(&CSFROM) VALUE(%SST(&REQUEST 33 4))
  CHGVAR VAR(&CST0) VALUE(%SST(&REQUEST 37 4))
  CHGVAR VAR(&CSCTYP) VALUE(%SST(&REQUEST 41 2))
ENDDO

/* Сервер баз данных */
CHGVAR VAR(&DBFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&DBFID) VALUE(%SST(&REQUEST 29 4))
/* Если формат равен ZDAD0100 */
IF COND(&CSFMT *EQ 'ZDAD0100') THEN(DO)
  CHGVAR VAR(&DBFILE) VALUE(%SST(&REQUEST 33 128))
  CHGVAR VAR(&DBLIB) VALUE(%SST(&REQUEST 161 10))
  CHGVAR VAR(&DBMBR) VALUE(%SST(&REQUEST 171 10))
  CHGVAR VAR(&DBAUT) VALUE(%SST(&REQUEST 181 10))
  CHGVAR VAR(&DBBFIL) VALUE(%SST(&REQUEST 191 128))
  CHGVAR VAR(&DBBLIB) VALUE(%SST(&REQUEST 319 10))
  CHGVAR VAR(&DBOFIL) VALUE(%SST(&REQUEST 329 10))
  CHGVAR VAR(&DBOLIB) VALUE(%SST(&REQUEST 339 10))
  CHGVAR VAR(&DBOMBR) VALUE(%SST(&REQUEST 349 10))
ENDDO

/* Если формат равен ZDAD0200 */
IF COND(&CSFMT *EQ 'ZDAD0200') THEN(DO)
  CHGVAR VAR(&DBNUM) VALUE(%SST(&REQUEST 33 4))
  CHGVAR VAR(&DBLIB2) VALUE(%SST(&REQUEST 37 10))
ENDDO

/* Если формат равен ZDAQ0100 */
IF COND(&CSFMT *EQ 'ZDAQ0100') THEN DO
  CHGVAR VAR(&DBSTMT) VALUE(%SST(&REQUEST 33 18))
  CHGVAR VAR(&DBCRRS) VALUE(%SST(&REQUEST 51 18))
  CHGVAR VAR(&DBSOPT) VALUE(%SST(&REQUEST 69 2))
  CHGVAR VAR(&DBATTR) VALUE(%SST(&REQUEST 71 2))
  CHGVAR VAR(&DBPKG) VALUE(%SST(&REQUEST 73 10))
  CHGVAR VAR(&DBPLIB) VALUE(%SST(&REQUEST 83 10))
  CHGVAR VAR(&DBDRDA) VALUE(%SST(&REQUEST 93 2))
  CHGVAR VAR(&DBCMT) VALUE(%SST(&REQUEST 95 1))
  CHGVAR VAR(&DBTEXT) VALUE(%SST(&REQUEST 96 512))
ENDDO

/* Если формат равен ZDAR0100 */
IF COND(&CSFMT *EQ 'ZDAR0100') THEN DO
  CHGVAR VAR(&DBLIBR) VALUE(%SST(&REQUEST 33 20))
  CHGVAR VAR(&DBRDBN) VALUE(%SST(&REQUEST 53 36))
  CHGVAR VAR(&DBPKGR) VALUE(%SST(&REQUEST 69 2))
  CHGVAR VAR(&DBATTR) VALUE(%SST(&REQUEST 89 20))
  CHGVAR VAR(&DBFULR) VALUE(%SST(&REQUEST 109 256))
  CHGVAR VAR(&DBMBRR) VALUE(%SST(&REQUEST 365 20))
  CHGVAR VAR(&DBFFMT) VALUE(%SST(&REQUEST 385 20))
ENDDO

/* Перечисленные ниже параметры относятся к формату ZDAR0200 */
/* Если формат равен ZDAR0200 */
IF COND(&CSFMT *EQ 'ZDAR0200') THEN DO
  CHGVAR VAR(&DBPLIB) VALUE(%SST(&REQUEST 33 10))
  CHGVAR VAR(&DBPTBL) VALUE(%SST(&REQUEST 43 128))

```

```

CHGVAR VAR(&DBFLIB) VALUE(%SST(&REQUEST 171 10))
CHGVAR VAR(&DBFTBL) VALUE(%SST(&REQUEST 181 128))
ENDDO

```

```

/* Сервер обработки удаленных команд */
CHGVAR VAR(&RCFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&RCFID) VALUE(%SST(&REQUEST 29 4))
CHGVAR VAR(&RCPGM) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&RCLIB) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&RCNUM) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&RCDATA) VALUE(%SST(&REQUEST 57 6000))

```

```

/* Объявление переменных для сервера входа в систему */
CHGVAR VAR(&SOFNT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&SOFID) VALUE(%SST(&REQUEST 29 4))

```

```

/*****/
/* */
/* Основная программа */
/* */

```

```

CHGVAR VAR(&STATUS) VALUE('1') /* Коду возврата присваивается значение, +
разрешающее прием запроса */

```

```

/* Общая часть для всех серверов */

```

```

/* Определение ID сервера */
IF COND(&APPLIC *EQ '*VPRT') THEN(GOTO CMDLBL(VPRT)) /* Если виртуальный принтер */
IF COND(&APPLIC *EQ '*TFRFCL') THEN(GOTO CMDLBL(TFR)) /* Если функция передачи данных */
IF COND(&APPLIC *EQ '*FILESRV') THEN(GOTO CMDLBL(FLR)) /* Если файловый сервер */
IF COND(&APPLIC *EQ '*MSGFCL') THEN(GOTO CMDLBL(MSG)) /* Если функция отправки сообщений */
IF COND(&APPLIC *EQ '*DQSRV') THEN(GOTO CMDLBL(DATAQ)) /* Если очередь данных */
IF COND(&APPLIC *EQ '*RQSRV') THEN(GOTO CMDLBL(RSQL)) /* Если удаленный запрос SQL */
IF COND(&APPLIC *EQ '*SQL') THEN(GOTO CMDLBL(SQLINIT)) /* Если запрос SQL */
IF COND(&APPLIC *EQ '*NDB') THEN(GOTO CMDLBL(NDB)) /* Если внутренний запрос к базе данных */
IF COND(&APPLIC *EQ '*SQLSRV') THEN(GOTO CMDLBL(SQLSRV)) /* Если SQL */
IF COND(&APPLIC *EQ '*RTVOBJINF') THEN(GOTO CMDLBL(RTVOBJ)) /* Если запрос на получение информации */
IF COND(&APPLIC *EQ '*DATAQSRV') THEN(GOTO CMDLBL(ODATAQ)) /* Если сервер очередей данных */
IF COND(&APPLIC *EQ '*QNPSRV') THEN(GOTO CMDLBL(NETPRT)) /* Если сервер сетевой печати */
IF COND(&APPLIC *EQ '*CNTRLSRV') THEN(GOTO CMDLBL(CENTRAL)) /* Если центральный сервер */
IF COND(&APPLIC *EQ '*RMTSRV') THEN(GOTO CMDLBL(RMTCMD)) /* Если сервер обработки удаленных команд */
IF COND(&APPLIC *EQ '*SIGNON') THEN(GOTO CMDLBL(SIGNON)) /* Если сервер входа в систему */

```

```

GOTO EXIT

```

```

/* * * * * * */
/* Функции */
/* */
/* * * * * * */

```

```

/* Виртуальный принтер */
VPRT:

```

```

/* Фрагмент кода */

```

```

GOTO EXIT

```

```

/* Функция передачи данных */
TFR:

```

```

/* Фрагмент кода */

```

```

GOTO EXIT

```

```

/* Файловый сервер */
FLR:

```

```

/* Фрагмент кода */

```

```

GOTO EXIT
/* Функция отправки сообщений */
MSG:

/* Фрагмент кода */

GOTO EXIT
/* Очереди данных */
DATAQ:

/* Фрагмент кода */

GOTO EXIT
/* Удаленные запросы SQL */
RSQL:

/* Фрагмент кода */

GOTO EXIT
/* Инициализация базы данных */
SQLINIT:

/* Фрагмент кода */

GOTO EXIT

/* Внутренние запросы к базе данных */
NDB:

/* Фрагмент кода */

GOTO EXIT
/* Запросы SQL */
SQLSRV:

/* Фрагмент кода */

GOTO EXIT
/* Получение информации об объекте */
RTVOBJ:

/* Фрагмент кода */

GOTO EXIT

/* Сервер очередей данных */
ODATAQ:

/* Фрагмент кода */

GOTO EXIT
/* Сервер сетевой печати */
NETPRT:

/* Фрагмент кода */

GOTO EXIT
/* Центральный сервер */
CENTRAL:

/* Фрагмент кода */

GOTO EXIT
/* Сервер обработки удаленных команд и вызовов распределенных программ */
RMTCMD:

```

```

/* В данном случае, если пользователь с идентификатором USERID отправит запрос на запуск */
/* на запуск удаленной команды или вызов распределенной программы, то запрос будет */
/* отклонен. */
IF COND(&USER *EQ 'userid') THEN(CHGVAR VAR(&STATUS) VALUE('0'))

        GOTO EXIT
/* Сервер входа в систему */
SIGNON:

/* Фрагмент кода */

GOTO EXIT

EXIT:
ENDPGM

```

Добавление новых функций в iSeries Access для Windows и Навигатор

В iSeries Access для Windows предусмотрена возможность изменения частей кода и встраивания новых функций. Такие функции называются дополнительными (или встраиваемыми) модулями. Их можно добавлять во время установки или обновления программы iSeries Access для Windows или рассылать пользователям во время выборочной установки. Для обновления установленных модулей служит функция Проверить уровень обслуживания.

Установка, удаление и обновление встраиваемых модулей

С помощью встраиваемых модулей можно добавлять новые функции и возможности в Навигатор iSeries. Такие функции представляют собой дополнительные компоненты, которые обычно добавляют:

- Папки и объекты в иерархический список
- Опции в меню Навигатора
- Страницы свойств к окнам свойств папок и объектов

Дополнительная информация о встраиваемых модулях и работе с ними приведена в разделе Создание встраиваемых модулей Навигатора.

Установка, удаление и обновление дополнительных модулей

Дополнительные модули - это удобное средство рассылки исходного кода по сети с помощью iSeries Access для Windows. Дополнительные модули могут содержать:

- Пользовательские программы
- Неупакованные файлы
- Программы установки и установочные образы продуктов

Для каждого дополнительного модуля должен быть создан файл ADDIN.INI с описанием модуля. Этот файл применяется программой установки iSeries Access для Windows, программой выборочной установки и программой проверки уровня обслуживания.

Примечание: Дополнительные модули - это простой и удобный способ рассылки файлов по сети. Однако при создании дополнительного модуля, содержащего приложения и программы установки, нужно учитывать некоторые требования и ограничения.

Работа со встраиваемыми модулями

Встраиваемые модули необходимо поместить в исходный каталог сервера. После этого их можно разослать пользователям во время обычной установки программы или выполнении процедуры выборочной установки. Установленные встраиваемые модули автоматически обновляются программой Проверить уровень обслуживания.

Дополнительная информация по этому вопросу приведена в разделе Рассылка встраиваемых модулей.

Установка и удаление встраиваемых модулей

Если встраиваемый модуль находится в источнике установки, он будет показан в списке компонентов Навигатора. Если встраиваемый модуль отсутствует в этом списке, его можно будет установить после завершения установки новой версии продукта, выполнив Выборочную установку. При запуске Выборочной установки укажите расположение нужного встраиваемого модуля (воспользуйтесь приведенной ниже таблицей). Программа выборочной установки выдаст список всех встраиваемых модулей в данном расположении, которые могут быть установлены. Однако если на клиенте и сервере установлены разные версии OS/400, то в списке могут быть не все дополнительные компоненты iSeries Access для Windows.

Встраиваемые модули iSeries Access для Windows расположены в следующих каталогах:

Встраиваемые модули	Расположение
IBM	(Имя AS/400 NetServer)\QIBM\ProdData\OpNavPlugin
Других фирм	(Имя AS/400 NetServer)\QIBM\UserData\OpNavPlugin

Встраиваемые модули для клиентов Client Access для Windows NT/95 расположены в следующих каталогах:

Встраиваемые модули	Расположение
IBM	(Имя AS/400 NetServer)\QIBM\ProdData\GUIPlugin
Других фирм	(Имя AS/400 NetServer)\QIBM\UserData\GUIPlugin

Примечание:

Если встраиваемый модуль не поддерживается продуктом iSeries Access для Windows напрямую, будет выдано предупреждающее сообщение. Тем не менее, этот модуль можно будет установить.

Обновление и обслуживание встраиваемых модулей

Для обновления встраиваемого модуля скопируйте обновленные файлы в каталог сервера, содержащий установочные образы встраиваемых модулей.

Программа Проверить уровень обслуживания автоматически обновит встраиваемый модуль. Эта программа определяет, какие встраиваемые модули требуется обновить, просматривая каталог сервера, содержащий установочные образы встраиваемых модулей. Если будут найдены модули, которые нужно обновить, программа проверки уровня обслуживания запустит функцию выборочной установки iSeries Access для Windows в специальном режиме. Эта функция обновит встраиваемый модуль.

Дополнительная информация об этом приведена в разделе Проверка уровня обслуживания

Работа с дополнительными модулями

Дополнительные модули можно устанавливать и удалять из системы во время установки продукта, его обновления или выборочной установки. Для обновления установленных дополнительных модулей служит функция Проверить уровень обслуживания. Файл ADDIN.INI. .

Установка и удаление дополнительных модулей

Для того чтобы установить дополнительный модуль вместе с iSeries Access для Windows, его нужно поместить в определенный каталог на сервере или в другой источник установки. Программа полной и выборочной установки iSeries Access для Windows выполняет поиск дополнительных модулей в следующем каталоге:

```
\QIBM\UserData\Ca400\Express\Addin\
```

При наличии нескольких дополнительных модулей каждый из них можно поместить в отдельный подкаталог.

Для установки дополнительного модуля выполните следующие действия:

1. Запустите программу установки iSeries Access для Windows или процедуру выборочной установки.
2. Заполните поля в окнах программы установки и выберите компоненты iSeries Access для Windows, которые вы планируете установить или удалить.
После установки или удаления компонентов iSeries Access для Windows появится окно "Установить дополнительные файлы и программы". В этом окне будет показан список дополнительных модулей, обнаруженных программой установки iSeries Access для Windows в соответствующем каталоге.
3. Рядом с каждым модулем показан переключатель. Отметьте переключатель, если модуль должен быть установлен на PC.
4. После выполнения оставшихся действий программа установки iSeries Access для Windows установит все выбранные дополнительные модули.

Для удаления дополнительного модуля запустите программу выборочной установки. После окна выбора компонентов появится список установленных дополнительных модулей. Выберите в списке те из них, которые нужно удалить из системы.

Примечание: Программа выборочной установки может удалить не все компоненты дополнительного модуля, если он содержит программы, которые сохраняют данные на компьютере, создают дополнительные файлы или добавляют записи в реестр. В таких случаях в дополнительный модуль нужно добавить программу, которую iSeries Access для Windows будет запускать перед удалением файлов дополнительного модуля. Инструкции по добавлению программ в дополнительные модули приведены в разделе ADDIN.INI.

Обновление и обслуживание дополнительных модулей

Для обновления дополнительного модуля скопируйте обновленные файлы в исходный каталог установки дополнительных модулей на хосте: \QIBM\UserData\Ca400\Express\Addin.

Программа Проверить уровень обслуживания автоматически обновит версию дополнительного модуля. Эта программа определяет, какие дополнительные модули нужно обновить, просматривая исходный каталог установки дополнительных модулей на хосте. Если будут найдены модули, которые нужно обновить, программа проверки версии запустит функцию выборочной установки iSeries Access для Windows в специальном режиме, и дополнительный модуль будет обновлен.

Дополнительная информация об этом приведена в разделе Проверка уровня обслуживания

Примечание: Для работы программы Проверить уровень обслуживания необходимо, чтобы в системе находился установочный образ дополнительного модуля.

Управление сервером iSeries NetServer

В iSeries Access для Windows используется стандартная функция OS/400 **Поддержка сетевого окружения Windows в системе OS/400 фирмы IBM (iSeries NetServer)**, предназначенная для управления файлами и печатью. Эта функция доступна начиная с версии 4, выпуска 2 OS/400. В старых версиях Client Access, например в Client Access для Windows 95/NT, были предусмотрены встроенные средства управления файлами и принтерами, что отражалось на цене продукта. Использование iSeries NetServer вместо встроенной поддержки iSeries Access для Windows дает следующие преимущества:

- Сокращается объем программы-клиента.
- Исчезает необходимость в фоновых задачах и демонах

С помощью iSeries NetServer в iSeries Access для Windows обеспечивается:

- Установка iSeries Access для Windows на PC из системы iSeries
- Управление файлами
- Управление печатью

Полная документация по работе с сервером iSeries NetServer приведена в разделе Сервер iSeries NetServer. Кроме того, ее можно просмотреть в главной панели навигации Information Center. Для этого откройте **Сеть > TCP/IP > iSeries NetServer**.

Ограничение возможностей пользователей с помощью стратегий и Администрирования приложений

iSeries Access для Windows поддерживает два основных средства управления сетью: Администрирование приложений и стратегии. Принцип администрирования приложений заключается в указании ограничений для пользовательского профайла iSeries с помощью Навигатора iSeries. Администрирование приложений поддерживается начиная с выпуска V4R3 OS/400; тем не менее, некоторые функции поддерживаются только начиная с выпуска V4R4 или более поздних. Стратегии задают параметры конфигурации и ограничения и применяются как к конкретным PC, так и к отдельным пользовательским профайлам Windows. Таким образом, стратегии являются более гибким, но и более сложным инструментом настройки, чем Администрирование приложений. Для применения стратегий необходимо загрузить Редактор системных стратегий Microsoft и настроить PC и сервер iSeries для хранения, получения и применения задаваемых стратегий. В общем случае, если Администрирование приложений позволяет ограничить доступ ко всем необходимым функциям и поддерживается текущей версией OS/400, то применять стратегии не рекомендуется.

В версии V5R2 в Администрировании приложений добавлена поддержка Общих параметров. Общие параметры позволяют управлять большей частью функций, для управления которыми в iSeries Access для Windows предусмотрены следующие шаблоны стратегий:

- Ограничения времени выполнения (caerestr.adm)
- Обязательные свойства соединения (config.adm)
- Стратегии настройки (caecfg.adm)

Дополнительная информация об Администрировании приложений и поддержке Общих параметров приведена в разделе Новое в версии V5R2 -Администрирование приложений.

Дополнительная информация об Администрировании приложений приведена в разделе Администрирование приложений.

Дополнительная информация о стратегиях приведена в следующих разделах:

- Обзор стратегий iSeries Access для Windows
- Настройка системы для применения стратегий
- Список стратегий iSeries Access для Windows

Обзор стратегий iSeries Access для Windows

С помощью стратегий вы можете запрещать пользователям выполнение некоторых действий, а также выдвигать рекомендуемые или обязательные требования к конфигурации. Стратегии применимы как к пользовательским профайлам Windows, так и к отдельным PC. Однако стратегии не позволяют управлять ресурсами сервера iSeries и не могут заменить средства защиты iSeries. Описание возможностей, которые дает применение стратегий, приведено в разделе Типы и области применения стратегий.

Поддержка стратегий при работе в сети

Стратегии хранятся на файловом сервере. При входе пользователя на рабочую станцию Windows последняя загружает все стратегии, относящиеся к этому пользователю. Стратегии применяются к реестру PC до того, как пользователь начнет работу. В операционных системах Windows 95 предусмотрена функция загрузки стратегий с сервера.

Для применения стратегий необходимы:

- Основной сервер входа в систему
- Сервер стратегий

В качестве сервера стратегий можно использовать Поддержку сетевого окружения Windows в iSeries фирмы IBM (AS/400 NetServer). Серверы Windows NT/2000 и Novell Netware могут выполнять обе эти функции.

Дополнительная информация приведена в разделе Настройка системы для применения стратегий.

Файлы стратегий

Определения стратегий содержатся в шаблонах стратегий. Шаблоны - это средство упорядочения стратегий по категориям. С iSeries Access для Windows поставляется пять шаблонов, по одному для каждой из следующих стратегий:

- Ограничение доступа к функциям iSeries Access для Windows для заданной системы (sysname.adm)
- Ограничение доступа к определенным функциям iSeries Access для Windows во время выполнения операций (caerestr.adm)
- Ограничение набора устанавливаемых/удаляемых компонентов для пользователей (caeinrst.adm)
- Установка рекомендуемых и обязательных значений параметров конфигурации для конкретных сред и систем в этих средах, а также некоторых настраиваемых значений для этих систем (config.adm)
- Установка рекомендуемых и обязательных глобальных настраиваемых значений (caecfg.adm)

Перед созданием или изменением стратегий создайте шаблоны стратегий с помощью утилиты CWBADGEN. Затем для активизации шаблонов и настройки соответствующих стратегий воспользуйтесь Редактором системных стратегий Microsoft. После создания стратегий сохраните изменения в файле стратегии, например (nt)config.pol.

Примечание: Стратегии для операционных систем Windows 95/98/Me и Windows NT/2000 следует создавать независимо друг от друга. (Стратегии для Windows 95 не работают в Windows NT.)

Дополнительная информация приведена в разделе Создание стратегий.

Типы стратегий и области их применения

Все стратегии iSeries Access для Windows подразделяются на стратегии ограничений и стратегии настройки. У каждой стратегии может быть как одна, так и несколько областей применения.

Стратегии ограничений

Для стратегий ограничений можно задавать любую область применения. Ниже перечислены возможные цели таких стратегий:

- Запретить или разрешить применение функции или действия iSeries Access для Windows.
- Задать ограничения на установку или удаление программных продуктов, компонентов, пакетов обслуживания и обновлений.
- Объединить несколько ограничений. Например, с помощью стратегии "Запрет на загрузку любых данных в iSeries" можно запретить некоторые или все типы передачи данных.
- Сделать скрытыми или недоступными определенные опции меню или элементы управления.
- Уведомлять пользователя о том, что выполнение выбранной им функции запрещено стратегией, путем выдачи сообщения на консоли или в окне.

Стратегии настройки

Стратегии настройки применимы только к отдельным пользователям. Ниже перечислены возможные цели таких стратегий:

- Заранее настраивать параметры конфигурации, необходимые пользователям.
- Настраивать параметры и режимы, которые пользователь обычно может включать или отключать, а также списки сред и соединений.
- Запрещать изменение обязательного значения. В этом случае поле с обязательным значением будет недоступно для изменения.

Стратегии настройки могут быть рекомендуемыми или обязательными.

- Рекомендуемые: Будут применяться значения параметров, заданные в стратегии, если они не задаются в явном виде пользователем или не устанавливаются приложением. Такой механизм позволяет переопределять стандартные значения, применяемые по умолчанию, при этом рекомендуемые значения не обязательно должны применяться - вместо них могут быть заданы другие значения.
- Обязательные: Будет применяться значение параметра, заданное в стратегии, - ни пользователь, ни прикладные программы не могут его изменить.

Области применения стратегий

Существует три возможных области применения стратегий: к компьютеру, к пользователю, к компьютеру и к соединению с системой iSeries. У одних стратегий может быть только одна область применения, у других - несколько.

Область применения	Описание
PC	Стратегия применима ко всем пользователям на данном PC. Исключение составляет случай, когда эта же стратегия применяется к конкретному пользователю для переопределения параметров, заданных для компьютера.

Область применения	Описание
Пользователь	Стратегия применяется к отдельным пользователям. Она может действовать только для некоторых пользователей и не действовать для других. Стратегия может быть установлена также и для "Пользователя по умолчанию" (любого пользователя без конкретной стратегии настройки). Некоторые стратегии, применяемые к пользователям, разрешают выполнять функции, запрещенные стратегиями, применяемыми ко всему компьютеру. При этом параметры конфигурации, установленные для этого компьютера, игнорируются.
Соединение с системой iSeries	<p>Некоторые стратегии, применяемые к пользователю или компьютеру, можно одновременно применять к соединению с системой iSeries. Если в качестве области применения стратегии указать соединение с iSeries, то параметры стратегии будут применяться только при работе с конкретной системой iSeries. Например, если указано, что стратегия ограничений должна применяться к пользователю USER1 и одновременно - к соединению с системой iSeries SYS1, то ограничения на функцию будут действовать только при работе пользователя USER1 с системой SYS1.</p> <p>Примечание: Если стратегия применяется к соединению с iSeries, то параметры стратегии для соединения имеют приоритет перед параметрами стратегии для пользователя или для компьютера. Например, если в качестве параметра, определяющего режим пользователя по умолчанию, для пользователя USER1 задано обязательное значение "Применять ИД пользователя по умолчанию", а для системы SYS1 - "Применять ИД и пароль пользователя Windows", то при подключении пользователя USER1 к системе SYS1 будет применяться его ИД и пароль Windows, а при его подключении к любой другой системе - ИД пользователя по умолчанию.</p> <p>Примечание: Для того чтобы применять стратегии к соединениям с iSeries, необходимо создать и использовать следующие шаблоны стратегий (один или оба):</p> <ul style="list-style-type: none"> • config.adm — шаблон настройки сред и соединений • sysname.adm — шаблон конкретной системы iSeries

Настройка системы для применения стратегий

Для применения стратегий iSeries Access для Windows выполните следующие действия:

1. Настройте сервер iSeries
2. Настройте компьютер-клиент для загрузки стратегий с сервера iSeries
3. Создайте файлы стратегий

Настройка сервера iSeries для работы со стратегиями

Перед работой со стратегиями выполните следующую процедуру настройки системы iSeries. Предполагается, что к сети подключены персональные компьютеры с операционной системой Windows.

- Настройте сервер iSeries в качестве iSeries NetServer, если это еще не сделано.
- Создайте папку интегрированной файловой системы для файлов стратегий.

Настройка PC для загрузки стратегий

Для того чтобы стратегии системы iSeries можно было загрузить на клиентские PC, последние необходимо настроить.

- Системы Windows 95/98/Me
- Системы Windows NT/2000/XP

Если поместить файл стратегии в общий каталог NETLOGON на сервере входа в систему iSeries 400, то PC пользователя будет автоматически загружать стратегию при входе пользователя в домен iSeries.

Настройка PC с Windows 95/98/Me для загрузки стратегий: Для загрузки и применения стратегий на вашем PC с Windows 95/98 выполните приведенную ниже процедуру.

1. Установите соединение между компьютером-клиентом и сервером iSeries NetServer по протоколу TCP/IP. Если в вашей сети применяется служба DNS, нужно занести имя сервера iSeries NetServer в таблицу хостов на сервере DNS. Если применяется файл LMHOSTS, то необходимо убедиться в том, что в нем содержится запись для iSeries NetServer с директивой #PRE, например:
9.4.3.240 QYOURSYS#PRE
2. Проверьте, установлено ли соединение между PC и системой iSeries.
3. Для того чтобы сервер мог применять стратегии для всех пользователей, включите функцию пользовательских профайлов на рабочем столе Windows.
 - a. Откройте **Пуск** → **Настройка** → **Панель управления** → **Пароль**.
 - b. Щелкните на вкладке **Профили пользователей**.
 - c. Убедитесь, что отмечен переключатель **Каждый пользователь устанавливает личные настройки, которые автоматически выбираются при входе в Windows**.
 - d. Нажмите **ОК** и перезагрузите компьютер.

Измените реестр таким образом, чтобы создаваемый вами файл стратегии можно было загрузить на каждый PC с Windows 95/98/Me в сети. Можно загрузить программу swbpoluz, которая будет делать это автоматически, со страницы: <http://www.ibm.com/eserver/iseries/access/cadownload.htm> .

Настройка PC с Windows NT/2000/XP для загрузки стратегий: В конфигурации всех компьютеров с Windows NT/2000/XP нужно задать опцию загрузки файла стратегий, который вы только что создали. Можно загрузить программу swbpoluz, которая будет делать это автоматически, со страницы:

<http://www.as400.ibm.com/clientaccess/cadownload.htm> .

Создание файла стратегий

Для создания или изменения стратегии нужно загрузить редактор стратегий фирмы Microsoft, создать шаблон стратегий, а затем создать или изменить файл стратегий.

1. Получить редактор стратегий.
2. Создать шаблоны файлов для iSeries Access для Windows.
3. Создать свои файлы стратегий.

Примечание: Стратегии для операционных систем Windows 95/98/Me и Windows NT/2000/XP следует создавать независимо друг от друга. (Стратегии, созданные для систем Windows 95/98/Me, не будут работать в системах Windows NT/2000/XP.)

Редактор стратегий фирмы Microsoft: Для создания собственных файлов стратегий необходим редактор стратегий, поставляемый фирмой Microsoft. Текущая версия редактора стратегий поставляется вместе с продуктами Windows NT Server, Windows NT Workstation Resource Kit и Office 97 Resource Kit. Кроме того, ее можно найти на Web-сайте Microsoft. В Windows 2000 предусмотрена собственная версия редактора стратегий, поставляемая вместе с Windows 2000 Server.

<http://www.microsoft.com> .

Найдите продукт **policy editor**. Устаревшая версия редактора стратегий поставляется на установочном компакт-диске для Windows 95. Не используйте ее. Эта версия позволяет загружать за один раз только один шаблон стратегии.

В соответствии с инструкциями по работе с редактором распакуйте файл и установите редактор стратегий и шаблоны.

Создание шаблонов стратегий iSeries Access для Windows: Вместе с iSeries Access для Windows поставляется программа для создания шаблонов стратегий, необходимых для управления стратегиями.

1. Откройте окно MS-DOS.
2. Перейдите в каталог iSeries Access для Windows. Обычно применяется следующий каталог:
[C:]\Program Files\IBM\Client Access\
3. Введите команду и параметры, необходимые для создания шаблонов стратегий, которые вы хотите настроить.

Команды создания шаблонов стратегий

Команда cwbadgen и параметры	Описание
<code>cwbadgen /ps S1034345</code> (где s1034345 - имя системы.)	Создает шаблон для установки стратегий для конкретной системы: S1034345.adm.
<code>cwbadgen /std</code>	Создает шаблоны caecfg.adm (управление глобальной конфигурацией), caeirst.adm (управление ограничениями на установку) и caerestr.adm (управление динамическими ограничениями).
<code>cwbadgen /cfg config.adm</code>	Создает шаблон config.adm (стратегия настройки на основе конфигураций системы, в которой была вызвана эта команда). Укажите имя файла шаблона в параметре /cfg. В данном примере создается файл шаблона config.adm.

Дополнительная информация об утилите cwbadgen приведена в руководстве по выполнению

Создание и обновление файла стратегий: Создайте файлы стратегии для управления действиями системы и пользователя по умолчанию.

1. Запустите редактор стратегий, дважды щелкнув на имени файла **poledit.exe**.
2. Выберите **Опции > Шаблон стратегии > Добавить**.
3. Перейдите в каталог, в котором хранятся созданные вами файлы шаблонов стратегий (с расширением .adm).
4. Выберите файлы .adm, которые нужно добавить, и нажмите кнопку **Добавить**. Повторите процедуру для всех нужных файлов .adm. Нажмите кнопку **ОК**.
5. Выберите **Файл > Создать стратегию**.
6. Настройте стратегии и сохраните файл стратегий в следующем каталоге:

\\QYOURSYS\POLICIES\config.pol (для Windows 95/98)

или:

\\QYOURSYS\POLICIES\ntconfig.pol (для Windows NT)

где

- QYOURSYS - имя вашего сервера AS/400 NetServer,
- POLICIES - имя папки с общими файлами на сервере AS/400 NetServer,
- (nt) config.pol - имя файла стратегий.

Для обновления файла стратегий откройте его в редакторе стратегий, внесите изменения и сохраните файл в том же каталоге.

Примечание: Для систем Windows 95/98/Me и Windows NT/2000 требуется создать разные наборы стратегий. (Стратегии, созданные для систем Windows 95, не работают в системах NT, и наоборот.)

Список стратегий iSeries Access для Windows

Программа iSeries Access для Windows поддерживает стратегии Microsoft System Policy. С помощью стратегий администраторы могут определять набор функций и параметров конфигурации, доступных отдельным пользователям. В данном разделе описаны все стратегии iSeries Access для Windows, их действие и область применения.

Набор стратегий определяется файлами шаблонов. На PC с iSeries Access для Windows можно создавать шаблоны стратегий для iSeries Access для Windows с помощью команды **cwbadgen**. Дополнительная информация приведена в разделе Создание шаблонов стратегий.

- Список стратегий,упорядоченный по их назначению
Упорядочивает список стратегий по их действию.
- Список стратегий,упорядоченный по шаблонам
Упорядочивает шаблоны и связанные с ними стратегии.

Общее описание стратегий iSeries Access для Windows приведено в разделе Обзор стратегий.

Список стратегий, упорядоченный по их назначению

Ниже приведен список стратегий iSeries Access для Windows, упорядоченный по их назначению.

Функция	Связанные с ней стратегии
Автообъекты ActiveX	<ul style="list-style-type: none"> • Запретить применение автообъекта для передачи данных • Запретить применение автообъекта для загрузки данных • Запретить применение автообъекта для работы с удаленными командами • Запретить применение автообъекта для работы с удаленными программами • Запретить применение автообъекта для работы с очередями данных
Связь	<ul style="list-style-type: none"> • Режим пользователя по умолчанию • Поиск адреса TCP/IP • Режим поиска порта • Обязательное применение SSL • Запрет на изменение активной среды • Запрет на изменение списка сред • Запрет на соединения с неопределенными системами • Запрет на использование необязательных сред • Тайм-аут соединения
Передача данных: Загрузка в систему iSeries	<ul style="list-style-type: none"> • Запрет на передачу любых данных в систему iSeries • Запрет на добавление информации в файлы сервера и на замену этих файлов • Запрет на загрузку данных с помощью GUI Передача данных • Запрет на использование RFROMPCB • Запрет на автоматическую передачу данных • Запрет на передачу надстроек Excel

Функция	Связанные с ней стратегии
Передача данных: Загрузка в систему iSeries	<ul style="list-style-type: none"> • Запрет на передачу данных из сервера iSeries • Запрет на загрузку данных с помощью GUI Передача данных • Запрет на использование RTOPCB • Запрет на автоматическую загрузку данных • Запрет на загрузку надстроек Excel
Передача данных: Создание файлов на сервере iSeries	<ul style="list-style-type: none"> • Запрет на создание файла хоста • Запрет на создание файлов на сервере iSeries с помощью мастера • Запрет на создание файлов на сервере iSeries без помощи мастера
Обновление каталога	Запрет на использование функции обновления каталогов
Удаленные команды	<ul style="list-style-type: none"> • Разрешение на выполнение любых удаленных команд, если отключено кэширование паролей • Выполнять от имени системы • Режим выполнения команд • Защита кэша • Разрешить общую защиту • Выполнять команды общей защиты от имени пользователя
Установка	<ul style="list-style-type: none"> • Исходный каталог выборочной установки • Запрет на установку • Запрет на выборочную установку • Запрет на удаление продукта из системы • Запрет на проверку уровня пакета обслуживания • Запрет на установку пакета обслуживания • Запрет на модернизацию • Запрет на перенос параметров iSeries Access для Windows из версий до V4R4M0 • Запрет на установку отдельных компонентов • Запрет на установку дополнительных модулей
Управление лицензиями	Время ожидания перед освобождением лицензии
Поддержка национальных языков	<ul style="list-style-type: none"> • Кодовая страница ANSI • Кодовая страница OEM • Кодовая страница EBCDIC • Двухнаправленное преобразование данных
ODBC	<ul style="list-style-type: none"> • Именованные источники данных • Запрет на применение источников данных, созданных программой
OLE DB	Запрет на применение комплекса связи OLE DB
Навигатор iSeries	Запрет на применение Навигатора iSeries

Функция	Связанные с ней стратегии
Пароли	<ul style="list-style-type: none"> • Предупреждать пользователя об истечении срока действия пароля iSeries • Разрешить кэширование паролей iSeries • Запрет на изменение паролей iSeries Access для Windows
Эмуляция PC5250	<ul style="list-style-type: none"> • Запрет на настройку сеансов дисплея • Запрет на настройку сеансов принтера • Запрет на использование эмулятора PC5250 • Максимальное число сеансов PC 5250 • Запрет на изменение профайлов .WS • Запрет на настройку меню • Запрет на настройку панели инструментов • Запрет на настройку нескольких сеансов • Запрет на настройку клавиатуры • Запрет на настройку мыши • Запрет на запуск апплетов Java • Запрет на вызов макрокоманд • Запрет на импорт профайлов в Диспетчере сеансов эмулятора • Запрет на удаление профайлов в Диспетчере сеансов эмулятора • Запрет на изменение каталогов в Диспетчере сеансов эмулятора
Команды PC	<ul style="list-style-type: none"> • Cwblogon • Cwbcfg • Cwbback • Cwbrest • Cwbenv • cwbundbs • cwbrxd • Wrkspfl • wrkmsg • wrkpri • wrkusri
Обслуживание	<ul style="list-style-type: none"> • Когда проверять • Время задержки • Частота • Скопировать образ на PC • Тихая установка • Исходный каталог установки • Автоматически запускать фоновое служебное задание
Пользовательский интерфейс	Запрет на создание значков на рабочем столе

Список стратегий, упорядоченный по шаблонам

Для управления стратегиями применяются перечисленные ниже файлы шаблонов. Дополнительная информация приведена в разделе создание шаблонов стратегий.

Файл шаблона	Описание
caecfg.adm	Стратегии, устанавливающие рекомендуемые или обязательные значения параметров конфигурации. Для создания введите команду cswbadgen с опцией /std.
caerestr.adm	Стратегии, устанавливающие ограничения на определенные функции iSeries Access для Windows. Для создания введите команду cswbadgen с опцией /std.
config.adm	Стратегии, устанавливающие обязательные значения параметров конфигурации для конкретных сред и систем в этих средах, а также некоторые настраиваемые значения в этих системах. Для создания введите команду cswbadgen с опцией /cfg.
caeinrst.adm	Стратегии, определяющие набор компонентов, которые пользователи могут устанавливать и удалять из системы, а также задающие ограничения на другие функции, связанные с установкой. Для создания введите команду cswbadgen с опцией /std.
SYSNAME.adm	Стратегии, устанавливающие ограничения на определенные функции iSeries Access для Windows для данной системы. Для создания введите команду cswbadgen с опцией /ps.

Администрирование Secure Sockets Layer

Secure Sockets Layer (SSL) - это широко распространенный протокол защиты, который позволяет компьютеру-клиенту идентифицировать сервер и зашифровывать все данные и запросы. Его рекомендуется применять при обмене конфиденциальной информацией с сервером. Типичный пример транзакций, защищаемых с помощью протокола SSL, - это операции по кредитным карточкам и передача номера банковского счета. В то же время, применение SSL повышает стоимость передачи данных из-за дополнительных операций по шифрованию и расшифровке.

В iSeries Access для Windows предусмотрен дополнительно устанавливаемый компонент - поддержка протокола Secure Sockets Layer (SSL), а также **Функция управления ключами IBM** для управления базами данных ключей. Протокол SSL поддерживают все функции iSeries Access для Windows, за исключением функций выполнения поступающих удаленных команд и Ultimedia. Однако на PC с 64-разрядным процессором Intel (например, Itanium), SSL может применяться только в 32-разрядных приложениях и соединениях. Продукт iSeries Access для Windows поддерживает связь SSL с сервером iSeries с 128-разрядным уровнем шифрования.

Начиная с версии V5R1, поддерживается идентификация клиентов в сеансах эмуляции PC5250.



Напечатано в Дании