IBM Redbooks: Creating Integrated Web Experience with IBM Lotus collaboration software

Note: This PDF document is the original text from the deliverable in the Lotus Sametime Wiki found in the URL in which this document originated. Always refer to the online wiki version for the latest updates.
# Contents

Chapter 1. Introduction to Creating Integrated Web Experience with IBM Lotus collaboration software .................................. 13
  1.1 Characteristics of a good integrated web experience .......... 13
  1.2 Purpose and scope of this guide .................................. 14
    1.2.1 Purpose .................................................. 14
    1.2.2 Scope .................................................... 14
    1.2.3 Out of scope ............................................ 14
  1.3 End product and day-in-the-life of a user ....................... 15
    1.3.1 Day-in-the-life of Philippe ............................... 15

Chapter 2. System architecture ............................................. 17

Chapter 3. Prerequisite information ....................................... 21
  3.1 Server configuration ........................................... 21
  3.2 DNS and host names ............................................. 21
  3.3 Directory paths .................................................. 22

Chapter 4. Installation of the integrated web environment servers ...................................................................................... 25
  4.1 Lotus Domino server installation ................................ 26
    4.1.1 Overview .................................................... 26
    4.1.2 Domino server installation ................................. 27
    4.1.3 Post-installation tasks ..................................... 28
    4.1.4 Verify Domino LDAP functionality ....................... 29
    4.1.5 Verify iNotes functionality ................................. 30
  4.2 IBM Sametime installation ........................................... 31
    4.2.1 Overview .................................................... 31
    4.2.2 Installation .................................................. 34
  4.3 IBM Connections installation ......................................... 53
    4.3.1 Overview .................................................... 53
    4.3.2 Prerequisite installation and configuration ......... 55
    4.3.3 Deploy Connections ....................................... 64
    4.3.4 Additional resources ....................................... 73
  4.4 Websphere Portal installation ......................................... 73
    4.4.1 Overview .................................................... 73
    4.4.2 Installation .................................................. 75
    4.4.3 Configuration ............................................... 76
    4.4.4 Installing fixpacks and fixes ............................. 78
    4.4.5 Verifying the install ....................................... 79
  4.5 Lotus Quickr for Domino installation .............................. 80
4.5.1 Overview ................................................................. 80
4.5.2 Installation ............................................................. 82
4.5.3 Configuration .......................................................... 87
4.5.4 Additional resources .................................................. 93

Chapter 5. Implement and test single sign-on ................................ 95
5.1 What is single sign-on .................................................... 95
  5.1.1 Single sign-on using LTPA ........................................ 95
5.2 Single sign-on architecture for the example environment .......... 96
5.3 Prerequisites for single sign-on ........................................ 96
5.4 Implementing single-sign on ............................................ 97
  5.4.1 Testing single sign-on ............................................. 105
  5.4.2 References ......................................................... 106

Chapter 6. Integration overview ............................................ 107

Chapter 7. Portal Integration ............................................... 109
7.1 Portal and iNotes integration ......................................... 109
  7.1.1 Overview ........................................................... 109
  7.1.2 Configuration on Portal ........................................... 109
  7.1.3 Configuration on Domino ........................................ 110
  7.1.4 Verifying the integration ......................................... 110
7.2 Portal and Connections integration .................................. 111
  7.2.1 Overview ........................................................... 111
  7.2.2 Prerequisites ........................................................ 111
  7.2.3 Required ifixes .................................................... 111
  7.2.4 Connection Portlets ............................................... 112
  7.2.5 Business card ........................................................ 118
  7.2.6 Verifying the integration ......................................... 119
  7.2.7 References ......................................................... 121
7.3 Portal and Quickr Domino integration ................................ 121
  7.3.1 Overview ........................................................... 121
  7.3.2 Configuration on Portal ........................................... 121
  7.3.3 Configuration on Domino ........................................ 122
  7.3.4 Verifying the integration ......................................... 123
  7.3.5 References ......................................................... 123
7.4 Portal and Sametime integration ...................................... 123
  7.4.1 Overview ........................................................... 124
  7.4.2 Prerequisites ........................................................ 124
  7.4.3 Installing Lotus Sametime proxy server upgrade for WebSphere Portal V7 ......................................................... 124
  7.4.4 Configuring an HTTP Server as a reverse proxy ............... 124
  7.4.5 Configuration on Sametime ....................................... 126
  7.4.6 Verifying the integration ......................................... 126
Chapter 8. Connections integration ........................................... 127
  8.1 Connections and Sametime integration ............................... 127
    8.1.1 Connections integration with STProxy for chat and awareness 128
  8.2 Connections and Quickr Domino integration ....................... 131
    8.2.1 Communities integration with Quickr Domino ................. 132
    8.2.2 Activities integration with Quickr Domino .................. 140

Chapter 9. iNotes integration ................................................. 145
  9.1 iNotes -- Sametime integration ..................................... 145
    9.1.1 Enable Sametime awareness in iNotes .......................... 145
    9.1.2 Testing Sametime awareness .................................... 147
  9.2 iNotes -- Quickr Domino integration .............................. 148
    9.2.1 Overview ....................................................... 148
    9.2.2 Create Policy settings to enable integration with Quickr . 148
    9.2.3 Enable Quickr preferences in iNotes .......................... 150
    9.2.4 Test Quickr integration ...................................... 151

Chapter 10. Quickr Domino integration .................................... 153
  10.1 Lotus Quickr for Domino and Sametime integration ............ 153
    10.1.1 Overview ..................................................... 153
    10.1.2 Preparing the Quickr and Sametime servers for awareness and Instant messaging .......... 154
    10.1.3 Configuring the Sametime server for Quickr Domino server . 154
    10.1.4 Verifying configuration ...................................... 155
    10.1.5 References .................................................... 156
  10.2 Lotus Quickr for Domino and Lotus Connections integration . 156
    10.2.1 Overview ..................................................... 156
    10.2.2 Configuring Quickr Domino to display photos from Connection profiles .... 157
    10.2.3 Configuring Quickr Domino to display Connections Business Cards . 159
    10.2.4 References .................................................... 160

Chapter 11. Sametime Integration ............................................. 161
  11.1 Overview of Sametime integration .................................. 161
  11.2 Sametime as a provider ............................................. 162
    11.2.1 Prepare Sametime to support instant messaging and presence awareness in Portal and Connections .................. 162
    11.2.2 Sametime as a consumer ...................................... 170
    11.2.3 Connections integration within Sametime .................... 173

Chapter 12. Creating common navigation across the products ......... 175
  12.1 Customizing the WebSphere Portal theme .......................... 176
    12.1.1 Overview ..................................................... 176

Contents 3
12.1.2 Changes to files within the Portal server PageBuilder2 directory . 177
12.1.3 PageBuilder2 changes within WebDAV directory .................. 200
12.1.4 References ................................................. 206
12.2 Customizing IBM Connections theme .............................................. 206
12.2.1 Customize the navigation bar ............................................. 207
12.2.2 Change logo to company image ........................................... 208
12.3 Customization of Lotus Quickr Domino theme .............................. 209
12.3.1 Overview ...................................................... 209
12.3.2 Customize the navigation bar in the Quickr Domino landing page 209
12.3.3 Customize the navigation bar in a Quickr Domino place .......... 215
12.3.4 Change logo to company image in the Quickr Domino Landing page 220
12.3.5 Change logo to company image in a Quickr Domino place ..... 221
12.4 Navigating from iNotes to other products ................................. 222
12.4.1 Overview ............................................. 222
12.4.2 Configuring an iNotes widget to open another web site ...... 222
Preface

IBM Lotus and IBM® Redbooks® have partnered together to create this wiki that show you how to use a specific set of IBM products to create an integrated web environment which provides a seamless, integrated web experience for users.

In this wiki, we describe the steps required to create an example integrated web environment based on the following IBM products. While doing this, we provide explanation of the steps, the choices we made, and some possible alternatives. The goal is to show you how you can create a similar environment for your deployment.

The products that this wiki covers (which our example environment uses) include:

IBM Lotus Domino - It is used to host critical business, collaboration, and messaging applications. In the example environment, the Domino server hosts the LDAP directory (Domino Directory) that all of the servers in the environment use for authentication. In addition, it hosts the iNotes application that supports web access to mail files stored on the server.

IBM Sametime: It provides users with a unified user experience across a broad range of integrated real-time communications services including voice, data and video. In the example environment, IBM Sametime provides instance messaging, presence awareness, online meetings, voice and video functions, and community collaboration features.

IBM Connections: It is the social software that helps you to find and connect with people and information in your organization. In the example environment, IBM Connections provides a set of integrated services including user profiles, files management, bookmarks, blogs, communities, wikis, forum, and activities management.

IBM WebSphere Portal: It provides the core portal services that aggregate applications and content and deliver them as role-based applications. The features and functionality IBM WebSphere Portal provides include flexible access to existing enterprise data and application as well as external feeds and web data, tagging and rating, REST services, site wizard, templates, theme customizer portlet, and WebSphere Portlet Factory software, Page Builder, and more.

IBM Lotus Quickr Domino: It is web-based collaboration software that helps you share content and collaborate with teams. The features IBM Lotus Quickr Domino provides include team places, content libraries, connectors, replacing
email attachment with shared links which connect team collaboration with other software, and more.

The wiki introduces the example environment we create, describes the system architecture, prerequisites, the installation and configuration of the above products, and at the end, how to create common navigation across the products to create the integrated web experience.

DISCLAIMER: IBM DOES NOT WARRANT OR REPRESENT THAT THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE PROVIDED ARE COMPLETE OR UP-TO-DATE. IBM DOES NOT WARRANT, REPRESENT OR IMPLY RELIABILITY, SERVICEABILITY OR FUNCTION OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE. IBM IS UNDER NO OBLIGATION TO UPDATE CONTENT NOR PROVIDE FURTHER SUPPORT.

ALL THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE ARE PROVIDED "AS IS," WITH NO WARRANTIES OR GUARANTEES WHATSOEVER. IBM EXPRESSLY DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ALL EXPRESS, IMPLIED, STATUTORY AND OTHER WARRANTIES, GUARANTEE ES, OR REPRESENTATIONS, INCLUDING, WITHOUT LIMITATION, THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF PROPRIETARY AND INTELLECTUAL PROPERTY RIGHTS. YOU UNDERSTAND AND AGREE THAT YOU USE THESE MATERIALS, INFORMATION, PRODUCTS, SOFTWARE, PROGRAMS, AND SERVICES, AT YOUR OWN DISCRETION AND RISK AND THAT YOU WILL BE SOLELY RESPONSIBLE FOR ANY DAMAGES THAT MAY RESULT, INCLUDING LOSS OF DATA OR DAMAGE TO YOUR COMPUTER SYSTEM.

IN NO EVENT WILL IBM BE LIABLE TO ANY PARTY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES OF ANY TYPE WHATSOEVER RELATED TO OR ARISING FROM USE OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE FOUND HEREIN, WITHOUT LIMITATION, ANY LOST PROFITS, BUSINESS INTERRUPTION, LOST SAVINGS, LOSS OF PROGRAMS OR OTHER DATA, EVEN IF IBM IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS EXCLUSION AND WAIVER OF LIABILITY APPLIES TO ALL CAUSES OF ACTION, WHETHER BASED ON CONTRACT, WARRANTY, TORT OR ANY OTHER LEGAL THEORIES.
The team who wrote this wiki

This wiki was produced by a team of technical specialists from around the world working together in remote fashion.

**Christina Cruz Born** is a Level 2 Software Engineer for IBM WebSphere Portal team. Christina joined IBM in 1997, first working as a Java developer and then moving to Level 2 Support in 2006. Christina holds certifications for WebSphere Portal development (7.0) and WebSphere Portal Administrator (6.0, 6.1, 7.0). She has a B.S. degree in Computer Engineering from the University of Florida. She is based in Atlanta, Georgia.

**Karen Brent** is an IT Specialist for IBM Software Group in United Kingdom. Karen has worked for Lotus and IBM for more than 12 years, initially within the Lotus services organization, where she assisted customers in designing, deploying, and managing Lotus Notes and Domino architectures. Currently, she is a Lotus Early Program Manager on the BetaWorks™ team, where she supports beta customers in deploying beta and early software, provides the development teams with feedback, and contributes to early enablement activities for the technical sales and services teams. Karen has been working with Lotus Notes and Domino since version 2.

**James Gallagher** is a Social Software Engineer currently working on IBM Connections and IBM Quickr for WebSphere Portal. James is a subject matter expert on both of these products and spends his time working directly with customers to solve technical issues that they may have. James has given customer presentations and workshops on the power of Social Software within today’s business. Within EMEA, he is the Senior Level 3 Software Engineer and point of contact for IBM Quickr for WebSphere Portal.

**Nagabhushan A Joshi (NJ)** is an IT Specialist in IBM India Software Lab. Nagabhushan has over 9 years of experience in the IT industry. He specializes in deployments, migrations, integration, and administration of IBM Lotus Domino and collaboration products. He was been involved in Lotus Notes and Domino infrastructure planning, implementation, and deployment projects in both technical and project management capacity. He is also an IBM Certified Lotus Domino Administrator. Nagabhushan currently works on middle-ware hosting services for IBM Websphere, DB2, and Lotus Domino environment.
Imran Khan is a Staff Software Engineer at IBM in Dublin Software Lab. He has more than 10 years of software development experience in IT industry. He is currently working as Media Manager product area expert and Sametime in general and is a member of EMEA Lotus Sametime Support Level 2 team. He holds a Masters degree in Mobile Networking and Computing from the University College Cork, Ireland. He is a Certified System Administrator for IBM Sametime and WebSphere Application Server. He is also a certified DB2 Associate. Imran's areas of expertise include large-scale IBM Sametime deployments and have keen interest in Unified Communications and Collaboration solutions and has also contributed to the writing of the Lotus Sametime 8.5 Enterprise Scale Deployment Redbooks wiki.

Feargal McKenna is a member of the Quickr-Domino (SEAL) team. He started working in Lotus development in 1994 as a technical specialist working on Lotus Domino/Notes. He has worked between 1999 and 2002 for Lotus Professional Services on Domino and QuickPlace and is now part of the Quickr Domino SEAL team since 2009 working with the development team and customers. Feargal has presented on Quickr Domino at Lotusphere, UKLUG, ILUG and BLUG. He has organized Master classes in Quickr Domino, developed education material for the Quickr Domino Design Partners, and he has lead Open Source projects in Quickr Domino.

Mark Neumann is an Executive Technical Sales Specialist and has been with IBM for 25 years. He works with customers on how to map IBM software to their business process. He has worked with IBM services, IBM development labs and customers all over the world on how to integrate IBM Collaboration products together with IBM and third party software. Mark has presented at past Lotusphere Impact conferences and IBM Portal Excellence conferences. Mark has created demonstrations and presentations showing the business value of the IBM Collaboration software products that are used worldwide by the IBM sales teams.

Powell Pendergraft is a Level 2 Software Engineer for IBM Lotus Domino Level 2 Webserver Support team, in IBM Software Group, US. Powell supports Domino HTTP, LDAP, DIIOP servers, iNotes, and Lotus Live Meetings. Over the last 12 years, Powell has worked with teams for cc:Mail, Domino mail servers, and the Domino Core support teams. Powell is based in Austin, Texas.
Charles Price is a Software Engineer in IBM Software Group, US. He has twelve years of experience in technical support and testing for IBM Lotus software, specializing in cross-product integration with Lotus and other products. Charles holds a degree in Mathematics Education from the University of Georgia and taught high school mathematics for three years before joining IBM. He is an IBM Certified Associate System Administrator - Lotus Collaborative Solutions, a Principal Certified Lotus Professional for Domino system administration, and an IBM Certified System Administrator for WebSphere® Portal. Charles has written numerous technotes on cross-product integration, presented at Lotus and Portal technical conferences, and co-authored the WebSphere Portal Collaboration Security Handbook, SG24-6438, in 2004 and Sametime 7.5.1 Best Practices for Enterprise Scale Deployment, SG24-7410, in 2007.

Wei-Dong (Jackie) Zhu is an Industry Solution Project Leader with International Technical Support Organization. She has more than 10 years of software development experience in accounting, image workflow processing, and digital media distribution using C, C++, Java, and Lotus Notes scripts. Jackie holds a Master of Science degree in Computer Science from the University of the Southern California. Jackie joined IBM in 1996. She is a Certified Solution Designer for IBM Content Manager and has managed and lead the production of many Enterprise Content Management Redbooks and IBM Lotus Domino redbooks wiki projects.

Supporting contributors and reviewers

Thanks to the following people for their contributions to this project:

- Jon Brun - Senior Software Engineer, IBM Connections
- Amanda Bauman - Business Program Manager, IBM Collaboration Solutions
- William Holmes - Technical Lead / Software Engineer, IBM Sametime Development
- Sven Stueven - Advisory IT Specialist, IBM WebSphere Portal Development
- Deepak Gupta1 - IT Specialist / SME - IBM WebSphere Portal
- Tony Estrada - Software Engineer, IBM Connections Development
- Projwal Saha - Software Engineer, IBM Connections Development
- Nagendra Shukla - Software Engineer, IBM Collaboration Solutions
- Richard Conway - ITSO Residency Enablement, IT Specialist
- Dave Bennin - ITSO Residency Enablement, IT Specialist
Become a contributor

Join us for a two- to six-week residency program! Share your knowledge with peers in the industry and learn from others. Help create content about specific products or solutions, while getting hands-on experience with leading-edge technologies. You will have the opportunity to team with IBM technical professionals, Business Partners, and Clients. Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you will develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:


Comments are welcome

Your comments are important to us!

We want the content in this wiki and all our wikis to be as helpful as possible. Provide us your comments in one of the following ways:

▶ Use the commenting feature with in the wiki. Login and add comments, located at the bottom of each page.
▶ Provide feedback in the Web form located at:

Stay connected to IBM Redbooks

▶ Find us on Facebook:
  http://www.facebook.com/IBMRedbooks
▶ Follow us on Twitter:
  http://twitter.com/ibmredbooks
▶ Look for us on LinkedIn:
  http://www.linkedin.com/groups?home=&gid=2130806
▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

Stay current on recent Redbooks publications with RSS Feeds:
http://www.redbooks.ibm.com/rss.html
Introduction to Creating Integrated Web Experience with IBM Lotus collaboration software

This section introduces this guide, explains the characteristics of a good integrated web experience, and how the products in the integrated environment can be used in a day-in-a-life of a user example.

1.1 Characteristics of a good integrated web experience

This guide discusses how to take a specific set of products (IBM Lotus Domino, IBM WebSphere Portal, IBM Lotus Quickr Domino, IBM Connections and IBM Sametime) and use them to create an integrated web environment. Before discussing how this is done, it is useful to consider what makes a good integrated web experience and what is the value that the integrated web environment brings to the user community.

A good integrated web experience is more than just a set of browser-based applications that are accessible to a user. It should have the following characteristics:

- Provide a single sign-on facility - Once users are logged into the integrated web environment, they should not be prompted to log in again.
- Be activity-centric - Allow users to focus on the business task that needs completing rather than the specific tools that are needed to complete it. For example, providing the ability to start, execute and complete a task from a single interface.
- Offer contextual collaboration - People and information related to the current task should be easily accessible. For example, having instant messaging capability and user profile information associated with all the places that user names occur.
- Use consistent navigation and menus - Make it easy for users to move seamlessly within and between applications. For example, all applications having the same set of menu options at the top of the screen.
Have a common look and feel - Make the experience aesthetically pleasing. The user interface should appear as a solution rather than a set of products. For example, each product including a company logo and using a common color theme.

The value of a good integrated web experience should be:

- Ease of use - Less clicks to get a job done.
- Efficient completion of tasks - People and information required to complete the tasks are easily available or contactable from within the tasks.
- Reduction in learning curve - Less training required on using the tools due to their similar look, feel and operation.

### 1.2 Purpose and scope of this guide

This section describes the purpose and scope of the guide, detailing what is and is not included in the documentation.

#### 1.2.1 Purpose

The purpose of this guide is to describe all of the steps required to create a specific example of an integrated web environment based on Domino, Sametime, Portal, Connections and Quickr Domino. While doing this, an explanation of the steps, the choices made and some possible alternatives will be discussed. The end result is intended to give guidance to companies on how they can create a similar environment using their own deployments which may not have exactly the same configuration, or contain all of the five collaboration products used in this example.

#### 1.2.2 Scope

The following items are in scope and are discussed in detail in the guide:

- Creation of a simple integrated example environment with single-server deployments of Domino, Connections, Portal and Quickr Domino and a two-server deployment of Sametime (one server for instant messaging services and one server for meetings).
- Structured step-by-step instructions to take stand-alone deployments of Domino/iNotes, Connections, Quickr Domino and Sametime and configure all the supported integration points available within the set of products.
- Instructions on how to customize the look and feel of the products to match the example environment.
- A video to demonstrate the use of the integrated environment in a day-in-the-life scenario.

#### 1.2.3 Out of scope

The following items are out of scope and are not discussed in detail in the guide. However, where appropriate, links are given to where information can be found on these topics:

- Detailed installation of the stand-alone products. The guide includes high-level detail on the installation steps involved in creating the example environment and focuses only on where non-default choices were made during the installations. Links are provided to where the detailed instructions for installing the products can be found.
Installation and configuration instructions for products outside of those detailed above. The example environment created and discussed in this guide could also be integrated with other IBM and non-IBM products (e.g. IBM Filenet P8 and Microsoft Outlook/Exchange) but these are not discussed in the guide.

Installation and configuration of production-level environments involving clustering, failover, scalability, security etc). The guide provides guidelines or links to information on these where they relate to configuring integration functionality.

1.3 End product and day-in-the-life of a user

A short description of a day-in-the life of a user helps to demonstrate the integration of the products and frequently used features. A video is also provided that goes along with the script.

1.3.1 Day-in-the-life of Philippe

Note: All the names of people used here are fictional and do not represent real people or companies.

Philippe Babineaux works in a geographically distributed team, He finds this integrated environment to be very useful. It enables him to work efficiently and closely with all team members and other corporate staff using the IBM Lotus and WebSphere servers and applications.

Here is an example of how he uses the software in the integrated environment to do his work.

Philippe opens his home page in WebSphere Portal and accesses his iNotes mail application. He opens an email from his manager who invites him to work on a project which will be a part of his company's community involvement project.

The email to Philippe (Phil) reads:

Hi, Phil,

The last week of March 2012 the General Area Children's Hospital, near our Pennsylvania headquarters, will host a Medieval/Renaissance weekend event. All entry fees and proceeds go to the fund raiser for a new hospital wing. I "volunteered" some of my fun loving staff to enter the competition as part of our community involvement program.

Your assignment, should you "accept" it, will be to design, build, and enter a scale model of an ancient siege engine called a hinged counter-weight trebuchet which can hurl a 30 lb. pumpkin 100 yards through a set of upright goal posts.

Let me know your plans, who you can put on a team, and what I can do to help make this happen.

Thanks,
Sunil

Using the integrated web environment, Phil performed the following tasks:

1. Phil opens the Applications tab of the Portal to access Connections to search for people located in the geographic area of the project, and for skills of different employees across geography so that the corporation on a wider basis can be involved. He looks for those
whose hobbies include history, mechanical engineering, building or carpentry, and anyone involved locally with the children’s hospital.

2. Phil finds the people and sends invitations for an impromptu Sametime meeting so they can discuss the project, including a link to his meeting room so the invitees can easily join the meeting.

3. The Sametime meeting attendees share their ideas and documents to help with the project direction.

4. Phil adds these documents to a Quickr Domino place he creates for the project.

5. Working through the Portal and Connections applications, Phil creates a community, adds members and begins an activity which describes and outlines the project. He then adds members, adds and assigns To Do’s to each member.

6. Some members comment and add documents to their To Do items showing their progress for the project.

7. One member adds an attachment to a Quickr place and sends links to other members which pictures some ideas found for the trebuchet.

8. Phil emails his manager, Sunil, with an update on the project and confirms the cost center code for the materials and the many pumpkins they need for testing.

DISCLAIMER: IBM DOES NOT WARRANT OR REPRESENT THAT THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE PROVIDED ARE COMPLETE OR UP-TO-DATE. IBM DOES NOT WARRANT, REPRESENT OR IMPLY RELIABILITY, SERVICEABILITY OR FUNCTION OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE. IBM IS UNDER NO OBLIGATION TO UPDATE CONTENT NOR PROVIDE FURTHER SUPPORT.

ALL THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE ARE PROVIDED “AS IS,” WITH NO WARRANTIES OR GUARANTEES WHATSOEVER. IBM EXPRESSLY DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ALL EXPRESS, IMPLIED, STATUTORY AND OTHER WARRANTIES, GUARANTEES, OR REPRESENTATIONS, INCLUDING, WITHOUT LIMITATION, THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF PROPRIETARY AND INTELLECTUAL PROPERTY RIGHTS. YOU UNDERSTAND AND AGREE THAT YOU USE THESE MATERIALS, INFORMATION, PRODUCTS, SOFTWARE, PROGRAMS, AND SERVICES, AT YOUR OWN DISCRETION AND RISK AND THAT YOU WILL BE SOLELY RESPONSIBLE FOR ANY DAMAGES THAT MAY RESULT, INCLUDING LOSS OF DATA OR DAMAGE TO YOUR COMPUTER SYSTEM.

IN NO EVENT WILL IBM BE LIABLE TO ANY PARTY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES OF ANY TYPE WHATSOEVER RELATED TO OR ARISING FROM USE OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE FOUND HEREIN, WITHOUT LIMITATION, ANY LOST PROFITS, BUSINESS INTERRUPTION, LOST SAVINGS, LOSS OF PROGRAMS OR OTHER DATA, EVEN IF IBM IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS EXCLUSION AND WAIVER OF LIABILITY APPLIES TO ALL CAUSES OF ACTION, WHETHER BASED ON CONTRACT, WARRANTY, TORT OR ANY OTHER LEGAL THEORIES.
System architecture

This section describes the system architecture and where product features are integrated within the environment.

In our integrated web environment, we deployed the following products:

1. IBM Lotus Domino
2. IBM WebSphere Portal
3. IBM Lotus Quickr for Domino
4. IBM Connections
5. IBM Sametime

For the first four products, each is deployed on its own server. Sametime is deployed on two servers, one for the instant messaging services and one for meetings.

The following diagram illustrates the products integrated in the example environment, from the perspective of a user accessing a specific server.
For example, if a user accesses the Portal server, then behind the scenes, the Portal server will be:

- Populating the iNotes portlet with data from the iNotes server.
- Displaying business cards for user names by pulling this information from the Connections server.
- Populating the Connections portlets with data from the Connections server.
- Populating the My Places portlet with data from the Quickr Domino server.
- Displaying presence awareness for user names by pulling this information from the Sametime server.
- Displaying instant messaging chat windows that are running on the Sametime server.

Another example, if a user accesses the Connections server, behind the scenes, the Connections server will be:

- Managing places hosted on the Quickr Domino server from within a Connections community.
- Adding files from activities to places on the Quickr Domino server.
- Displaying presence awareness for user names by pulling this information from the Sametime server.
- Displaying instant messaging chat windows that are running on the Sametime server.

This diagram provides the reference point throughout the wiki to indicate the servers and integration components involved in a specific piece of configuration.
DISCLAIMER: IBM DOES NOT WARRANT OR REPRESENT THAT THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE PROVIDED ARE COMPLETE OR UP-TO-DATE. IBM DOES NOT WARRANT, REPRESENT OR IMPLY RELIABILITY, SERVICEABILITY OR FUNCTION OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE. IBM IS UNDER NO OBLIGATION TO UPDATE CONTENT NOR PROVIDE FURTHER SUPPORT.

ALL THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE ARE PROVIDED "AS IS," WITH NO WARRANTIES OR GUARANTEE WHATSOEVER. IBM EXPRESSLY DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ALL EXPRESS, IMPLIED, STATUTORY AND OTHER WARRANTIES, GUARANTEE, OR REPRESENTATIONS, INCLUDING, WITHOUT LIMITATION, THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF PROPRIETARY AND INTELLECTUAL PROPERTY RIGHTS. YOU UNDERSTAND AND AGREE THAT YOU USE THESE MATERIALS, INFORMATION, PRODUCTS, SOFTWARE, PROGRAMS, AND SERVICES, AT YOUR OWN DISCRETION AND RISK AND THAT YOU WILL BE SOLELY RESPONSIBLE FOR ANY DAMAGES THAT MAY RESULT, INCLUDING LOSS OF DATA OR DAMAGE TO YOUR COMPUTER SYSTEM.

IN NO EVENT WILL IBM BE LIABLE TO ANY PARTY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES OF ANY TYPE WHATSOEVER RELATED TO OR ARISING FROM USE OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE FOUND HEREIN, WITHOUT LIMITATION, ANY LOST PROFITS, BUSINESS INTERRUPTION, LOST SAVINGS, LOSS OF PROGRAMS OR OTHER DATA, EVEN IF IBM IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS EXCLUSION AND WAIVER OF LIABILITY APPLIES TO ALL CAUSES OF ACTION, WHETHER BASED ON CONTRACT, WARRANTY, TORT OR ANY OTHER LEGAL THEORIES.
Prerequisite information

Before the example environment is built, certain configuration information has to be agreed so that the servers can integrate with each other easily. This section describes the prerequisites for setting up the integrated environment.

3.1 Server configuration

For consistency and simplicity, the specification for all of the servers in the example environment is the same:

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard Disk</td>
<td>100Gb</td>
</tr>
<tr>
<td>Memory</td>
<td>12Gb</td>
</tr>
<tr>
<td>Processor</td>
<td>Dual Core</td>
</tr>
<tr>
<td>Operating System</td>
<td>Windows Server 2008 R2 64 bit</td>
</tr>
</tbody>
</table>

The hard disk is divided into three separate logical disks each with a specific purpose:

- C:\ (40Gb) contains the operating system.
- E:\ (40Gb) contains the installed programs.
- F:\ (20Gb) contains the install kits.

3.2 DNS and host names

In order for single sign-on to work across the integrated environment, the servers are all configured as part of the same domain demos.ibm.com

To make it simple to identify the servers, each server is given a host name that reflects the main product that is installed. In addition, since there is more than one Sametime service running on each of the Sametime servers, aliases are assigned to each Sametime service so that it is easy to distinguish between them in the configuration.
Server names and functions are displayed in the following table:

<table>
<thead>
<tr>
<th>Hostname</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>domino.demos.ibm.com</td>
<td>Domino LDAP and iNotes server</td>
</tr>
<tr>
<td>portal.demos.ibm.com</td>
<td>Portal server</td>
</tr>
<tr>
<td>quicker.demos.ibm.com</td>
<td>Quicker Domino server</td>
</tr>
<tr>
<td>connections.demos.ibm.com</td>
<td>Connections server</td>
</tr>
<tr>
<td>sametime.demos.ibm.com</td>
<td>Sametime Community server</td>
</tr>
<tr>
<td></td>
<td>Sametime Proxy server</td>
</tr>
<tr>
<td></td>
<td>Sametime Console server</td>
</tr>
<tr>
<td>collab.demos.ibm.com</td>
<td>Sametime Meeting server</td>
</tr>
<tr>
<td></td>
<td>Sametime Media server</td>
</tr>
</tbody>
</table>

Because the servers are not registered with a DNS server, these host names are configured in the Computer properties (Control Panel > System and Security > System) as shown below.

```
Computer name, domain, and workgroup settings

Computer name: connections
Full computer name: connections.demos.ibm.com
Computer description: 
Workgroup: WORKGROUP
```

An entry is added for each server in the hosts file (C:\Windows\system32\drivers\etc\hosts) on each computer so that the servers can locate and communicate with each other.

### 3.3 Directory paths

We recommend that the installation directory in all of the installations is edited to remove “Program Files" and any other additional directory names from the path that do not add value to the structure or organization of the program.

For example:

"\Program Files\IBM\SQLLIB" becomes "\IBM\SQLLIB"

or

"\Program Files\IBM\Lotus\Domino" becomes "\Lotus\Domino"

Reasons for this recommendation:

- There are situations where spaces in path names can cause issues.
- Shortening the directory name can avoid issues where the installation path exceeds the Windows limitations for path length.
Chapter 4. Installation of the integrated web environment servers

This section provides high level details on the installation of the servers that make up the integrated web environment to a level where the integration functionality between the various servers can be configured.

It includes the following subsections:

- 4.1, “Lotus Domino server installation” on page 26
- 4.2, “IBM Sametime installation” on page 31
- 4.3, “IBM Connections installation” on page 53
- 4.4, “Websphere Portal installation” on page 73
- 4.5, “Lotus Quickr for Domino installation” on page 80
4.1 Lotus Domino server installation

This section describes how to install the Domino server to the level used in the integration environment.

4.1.1 Overview

IBM Lotus Domino is used to host critical business, collaboration, and messaging applications. It delivers reliable, scalable, and secure applications, helping companies enhance the productivity of people, streamline business processes and improve overall business responsiveness.

In the example environment the Domino server has a dual purpose.

- It hosts the LDAP directory (Domino Directory) that all of the servers in the environment use for authentication.
- It hosts the iNotes application that supports web access to mail files stored on the server.

Server diagram

Prerequisites

The Domino server will act as the user registration server, the mail server using Domino HTTP for access to iNotes based mail files, and the directory server using Domino LDAP. Use the guide to deployment to help you survey your environment and needs.

When you plan certifiers, host names, and user Distinguished Name formats consider what use will be made of the Notes hierarchy and how the naming scheme for the certifiers,
servers, and user names will reflect your organization. Be sure to use the guide to naming conventions and standards during this stage. You can, also, access the online Domino administrator help guide.

In the example environment, the certifier name is Demos. In Chapter 2, “System architecture” on page 17, you can see how the servers are related to each other and their planned purposes by the name of the certifier and names of the servers.

For information on the basic hardware and operating system configuration of the server, refer to Chapter 3, “Prerequisite information” on page 21.

For more information on specific software and hardware requirements for Domino, please refer to the following technote:

IBM Lotus Domino 8.5.3 is installed in the example environment. The following is the complete list of products and part numbers required for this specific installation.

<table>
<thead>
<tr>
<th>Part numbers</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1L10EN</td>
<td>IBM Lotus Domino V8.5.3 64 bit for Windows English</td>
</tr>
<tr>
<td></td>
<td>Lotus Notes 8.5.3 IF1</td>
</tr>
<tr>
<td>C1L0EN</td>
<td>IBM Lotus Notes, Domino Designer and Admin V8.5.3 for Windows XP, Vista and Windows 7 32 Bit English</td>
</tr>
</tbody>
</table>

### 4.1.2 Domino server installation

You can read the full details and considerations for installing the Domino server in the IBM Lotus Domino and Notes Information Center. Details below illustrate items of particular interest during the installation process.

1. The Domino server is installed using all the default options.
2. In the Domino setup the following non-default choices are made:

<table>
<thead>
<tr>
<th>Option</th>
<th>Choice</th>
</tr>
</thead>
<tbody>
<tr>
<td>First or additional server</td>
<td>Setup first server...</td>
</tr>
<tr>
<td>Server name</td>
<td>domino</td>
</tr>
<tr>
<td>Organization name</td>
<td>domos</td>
</tr>
<tr>
<td>Domain name</td>
<td>domos</td>
</tr>
<tr>
<td>Administrator name</td>
<td>Domino Admin</td>
</tr>
<tr>
<td>Internet services</td>
<td>Select Web Browsers (HTTP services), Internet Mail Clients, Directory services (LDAP services)</td>
</tr>
<tr>
<td></td>
<td>• HTTP is required for hosting iNotes</td>
</tr>
<tr>
<td></td>
<td>• Internet Mail Clients are required so that other services can send notifications to users via SMTP</td>
</tr>
<tr>
<td></td>
<td>• LDAP is required for hosting the common LDAP directory for the example environment</td>
</tr>
<tr>
<td>Host name</td>
<td>domino.demos.ibm.com</td>
</tr>
</tbody>
</table>

**Verify Domino installation**

Perform the following steps to verify the Domino installation:
1. Start the Domino server and check the console to see that the server is started.
2. From a command prompt type:
   
telnet host name of the primary Domino Server 1352

   The result should be a blank Telnet window:

3. From a browser type:
   
http://Domino host name/names.nsf

   and log in with your administrator credentials. The Person view of the Domino directory should be displayed.

4.1.3 Post-installation tasks

This section details the tasks carried out once the Domino server has been installed.

**Apply Lotus iNotes 8.5.3 IF1**

Perform the following steps to install the fix pack:

1. Shut down the Domino server.
2. In the Domino Data/iNotes directory, rename Forms85.nsf to Forms85.nsfOEM853.
3. Copy the new Forms85.nsf from the fix pack into the Domino Data/iNotes directory.
4. Restart the Domino server.

**Install Domino Administrator client**

Once the server has been installed, Notes clients can be installed. For the example environment a single "All Clients" installation is performed on the Domino server to provide a Domino administrator client that is used to administer the server. For detailed information on installing the client refer to the following document:
To validate the installation:
1. Start the Domino Administrator client.
2. Log in with your Domino administrator credentials.
3. Open the Domino directory (names.nsf) on the server.

**Domino user registration**

You may register users, including Notes IDs and mail files, by entering them one at a time in the Register Person dialog. More efficient and much more simple to modify on an as needed basis, you may use a text file with attributes, ID, and mail file for each user defined, with one user per line of text.

For the example environment:
1. Around 80 users are registered from a pre-populated text file.
2. Once the users are registered, additional information is added to each user record to flesh out the user characters. This information includes phone numbers, office locations, job roles, manager and secretary names.

---

**Person: Amadou Alain/demos**

<table>
<thead>
<tr>
<th>Work/Home</th>
<th>Other</th>
<th>Miscellaneous</th>
<th>Certificates</th>
<th>Process</th>
<th>Administration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Work</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Home</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Corporate Hierarchy Information</td>
</tr>
</tbody>
</table>

**Work**
- **Title:** Marketing Manager
- **Company:** Demos
- **Department:** 2M2
- **Employee ID:** 14332
- **Location:** Chicago
- **Manager:** Simone Drey/demos

**Company**
- **Street address:** 10352 International Drive
- **City:** Chicago
- **State/Province:** Illinois
- **Zip/postal code:** 60603
- **Country:** us
- **Office number:** ch

**Office phone:** 1-301-555-8100
**FAX phone:** 301-555-3112
**Cell phone:** 1-312-555-3276
**Pager number:**
**Assistant:** Rita Ferrey/demos

---

**4.1.4 Verify Domino LDAP functionality**

To validate that the Domino LDAP is functioning:
1. Open a command prompt on the server.
2. Change to the Notes client program directory.
3. Type the following at the command prompt.

```
ldapsearch -h Domino server hostname cn=user common name
```
where *user common name* is the name of a user in the Domino directory.

If the details of the appropriate user are returned, the LDAP task on the Domino server is working correctly.

For more information on configuring and testing the LDAP task refer to the following document:


### 4.1.5 Verify iNotes functionality

To validate that Lotus iNotes is working correctly.

1. Type the following URL in a browser window.

   http://Domino server hostname/mail/user mail file

   where *user mail file* is the file name of one of the mail files on the server

2. Log in as the owner of the mail file and the iNotes Home page should be displayed.
For more information on configuring and testing the HTTP task refer to:


4.2 IBM Sametime installation

This section describes Sametime installation and configuration for stand alone usage.

4.2.1 Overview

IBM Sametime provides users with a unified user experience across a broad range of integrated real-time communications services including voice, data and video. It also provides a unified communications and collaboration platform, with built-in integration with many productivity and collaborative applications, and with open-standards-based tools to help developers provide Sametime services as part of business processes.

The features and functionality of IBM Sametime include:

► **Instant Messaging:** Quick text chats that can answer simple questions outright or can be escalated to multi-way voice or video chats or online meetings.

► **Presence Awareness:** Indicates whether now is a good time to initiate a real-time conversation via instant messaging or a phone/conference call.

► **Online Meetings:** Supports collaboration which requires document, application and/or screen sharing.

► **Voice and Video:** Provides high-quality integrated desktop voice and video to help cut telephony costs.
- **Telephony**: Integrates telephony and provides telephony features to support a unified communications strategy.
- **Community Collaboration**: Tools that help people find, reach and collaborate with communities of users who may not be in their contact lists.

There are three different Sametime offerings:

- Entry
- Standard
- Advanced

Each of the above offerings has different features. For more information, please see the following documentation:


Any of the Sametime offerings could be used in the example environment. Sametime Standard 8.5.2 is used because it is the most comprehensive offering with respect to features that can be integrated with the other products.

**Server diagram**

The example environment built for this project includes a dual-server Sametime deployment. In a production environment, it is very likely that the services described below would be spread over a number of servers and duplicated for performance, scalability and high availability purposes.

One of the physical servers, designated as the Sametime Messaging server, hosts the Sametime 8.5.2 Community server, Sametime 8.5.2 System Console server and Sametime 8.5.2 Proxy server, and the other, designated as the Sametime Meetings server hosts the Sametime 8.5.2 Meeting server.

The Sametime Messaging server has a DB2 data store which is used by two of the Sametime servers (System Console and Meeting Server) to store the data associated with their services.

The Community server is deployed on a Domino server and the System Console, Proxy and Meeting servers are deployed on WebSphere Application server in a deployment manager and cell configuration (spanning both physical servers), with a node agent to facilitate communication and synchronization between the deployment manager and the other servers in the cell. The System Console is used to manage all of the servers, including the Domino-based Community server, and an LDAP hosted on the iNotes (Domino) server is used for authentication.
The dotted lines in the diagram indicate the communication paths used for configuring the system.

The dashed lines in the diagram indicate the communication paths used for authentication.

The unbroken lines in the diagram indicate the communication paths used in operating the system.

Note: It would be possible to install Sametime 8.5.2 Media Manager within this environment but there are no specific integration capabilities between Sametime Media Manager and the other portfolio products in the example environment so this is not covered in this wiki.

Note: Additional installation and configuration are required to support the surfacing of instant messaging and awareness within Portal and Connections. These are covered in Chapter 11, “Sametime Integration” on page 161.

**Prerequisites**

For information on the basic hardware and operating system configuration of the server, refer to Chapter 3, “Prerequisite information” on page 21.

For more information on software and hardware requirements, please refer to the following technote:


Download the following part numbers to install all Sametime components:
For more information on downloading the Sametime install images, please refer to the following technote:


**Hosts file**

The Sametime services are spread across two physical server machines so, to make it clear which service is being configured at any one time, separate aliases are set in the hosts file so that the services can be distinguished from one another.

<table>
<thead>
<tr>
<th>Physical server Host Name</th>
<th>Service</th>
<th>Host Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>sametime.demos.ibm.com</td>
<td>Sametime Community server</td>
<td>sametime.demos.ibm.com</td>
</tr>
<tr>
<td></td>
<td>Sametime System Console</td>
<td>ssc.demos.ibm.com</td>
</tr>
<tr>
<td></td>
<td>Sametime Proxy server</td>
<td>webchat.demos.ibm.com</td>
</tr>
<tr>
<td>collab.demos.ibm.com</td>
<td>Sametime Meeting server</td>
<td>meeting.demos.ibm.com</td>
</tr>
</tbody>
</table>

### 4.2.2 Installation

This section describes, at a high level, the steps required to install the Sametime servers that host the Sametime services available in the example environment, with any non-default choices that are made. Where appropriate, links to more detailed information on these steps are provided.

**Domino Server**

This section describes the installation of the Domino server that will host the Sametime Community server.

1. The Domino 8.5.2 server is installed using all the default options. For more detailed information please refer to the following document:
   
   http://www-10.lotus.com/ldd/stwiki.nsf/dx/Installing_a_Lotus_Domino_server_st85_2

2. In the Domino setup the following choices are made:
3. Domino 8.5.2 fix pack 3 is installed on the Domino server using all the default options.

**DB2 Server**
The System Console uses DB2 to store store configuration information on each of the Sametime components that it manages. The Meeting server uses DB2 to store information on each of the meeting rooms that it hosts.

This section gives an overview of the high level steps involved to install DB2 9.7. For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Installing_DB2_on_Linux_or_Windows_st852

1. For the example environment, the default install options are selected with the exception of the following:
   - Notifications are disabled (In a production environment you might choose to have notifications sent to a specific administrator account).

2. To validate the installation, check the system tray. If DB2 has been installed successfully it is started and the **DB2 Control Center** icon is displayed in the system tray.

**Sametime System Console**
The Sametime System Console is used to configure and manage the Sametime deployment. This section details the installation of the System Console server.

### Create System Console database
This section details the creation of the System Console database.

1. First the database that will hold all the configuration information is created in DB2. For detailed information on this task please refer to the following document:

   http://www-10.lotus.com/ldd/stwiki.nsf/dx/Creating_a_database_for_the_system_console_on_AIX_Linux_Solaris_or_Windows_st852
2. To validate that the database has been created, open the DB2 command center. If the STSC database is displayed, the installation is successful.

![DB2 Command Center]

**Install System Console**

In this section the Sametime System Console server is installed. For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Installing_the_console_on_AIX_Linux_Solaris_or_Windows_st852

1. For the example environment, the default install options were selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>WAS Configuration</td>
<td>HostName</td>
<td>ssc.demos.ibm.com</td>
<td>Used to distinguish between the various services that run on sametime.demos.ibm.com</td>
</tr>
<tr>
<td>DB2 Configuration</td>
<td>HostName</td>
<td>ssc.demos.ibm.com</td>
<td></td>
</tr>
</tbody>
</table>

2. To validate the install, launch the Integrated Solutions Console and if the Sametime System Console has an entry in the navigator, the install has completed successfully.

![Integrated Solutions Console]

**Configure LDAP**

In this section the Sametime System Console server is configured to connect to the LDAP server using the Integrated Solutions Console. For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Connecting_to_an_LDAP_server_st852
1. For the example environment, the default install options are selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bind to LDAP</td>
<td>Deployment Name</td>
<td>domino_ldap</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Host Name</td>
<td>domino.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bind distinguished name</td>
<td>cni=wpsbind</td>
<td>ID created in the LDAP for the purpose of allowing authenticated LDAP binds.</td>
</tr>
<tr>
<td>Connect to LDAP Servers</td>
<td>LDAP base entry</td>
<td>o=demos</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Configure advanced LDAP settings</td>
<td>check this box</td>
<td></td>
</tr>
<tr>
<td>Group attributes</td>
<td>LDAP group search base</td>
<td>blank</td>
<td>Lotus Domino uses a flat naming convention for its groups. Therefore it is important to ensure that the value specified here is blank so that groups can be resolved in Sametime.</td>
</tr>
</tbody>
</table>

2. After the installation, stop and start the deployment manager.

3. To validate the configuration, launch the Integrated Solutions Console and select Users and Groups > Manage Users and click Search with a search target of "*". If the list of users from the LDAP are returned, the configuration is successful.

88 users matched the search criteria.
**Sametime Community server**
The Sametime Community server provides the instant messaging services for the Sametime deployment. This section describes the steps to install the Sametime Community server.

**Create Sametime Community server deployment plan**
In this section the Sametime System Console is used to create a deployment plan for the Sametime Community server. For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Guided_activity_Preparing_to_install_a_Sametime_Community_Server_st852

1. Select Install Sametime Community Server from the Sametime Guided Activities in the Integrated Solutions Console. For the example environment, the default options were selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Sametime Server</td>
<td>Deployment Name</td>
<td>stdomino</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Product Version</td>
<td>8.5.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Host Name</td>
<td>sametime.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Domino User ID</td>
<td>domino admin</td>
<td>Administrator of the Domino server</td>
</tr>
<tr>
<td>Connect to an LDAP server</td>
<td>Deployment Name</td>
<td>domino_ldap</td>
<td></td>
</tr>
</tbody>
</table>

**Install Sametime on Domino server**
In this section, the Sametime Community Server is installed on the Domino server. For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Installing_a_community_server_on_AIX_Linux_Solaris_or_Windows_st852

1. Stop the Domino server (on the Sametime server) before starting the Sametime Community Server installation.

2. For the example environment, the default install options are selected with the exception of the following:
3. When the installation is complete, start the Domino server.

4. To verify the installation type the following URL into a browser, login as the Domino administrator and select Administer the server. If all the services in the list, except for "Telephony Services" have a status of "Running" the Sametime server is operational.

Sametime Proxy server

In the example environment, Sametime will provide presence awareness and chat services for all of the other products. This section describes the steps to install the Sametime Proxy server.

For detailed information on installing a Sametime 8.5.2 proxy server please refer to the following document:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Installing_a_Sametime_Proxy_Server_st852

Create Deployment Plan

In this section the Sametime System Console is used to create a deployment plan for the Sametime Proxy server.

2. For the example environment, the default options are selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Sametime Proxy Server</td>
<td>Deployment Name</td>
<td>852_proxy</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Product Version</td>
<td>8.5.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Configuration type</td>
<td>Primary Node</td>
<td></td>
</tr>
<tr>
<td>Node Federation</td>
<td>System Console:sametimeSSCCell</td>
<td></td>
<td>The Proxy server will be part of the System Console cell</td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>Host Name</td>
<td>webchat.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td></td>
<td>User ID</td>
<td>was.admin</td>
<td></td>
</tr>
<tr>
<td>Connect to Communities server</td>
<td>Deployment Name</td>
<td>stdomain</td>
<td></td>
</tr>
</tbody>
</table>

**Install Sametime Proxy server**

In this section, the Sametime Proxy Server is installed on the same server as the System Console and the Sametime Community server.

1. For the example environment, the default install options are selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Packages</td>
<td>Package Group</td>
<td>Use existing</td>
<td>Existing WebSphere Application Server installation will be used rather than installing an additional one alongside the existing.</td>
</tr>
<tr>
<td></td>
<td>Use System Console to install</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WAS Location</td>
<td>Host name (SSC)</td>
<td>ssc.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Host name (proxy)</td>
<td>webchat.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td>Deployment Plan List</td>
<td>Dep Name</td>
<td>852_proxy</td>
<td></td>
</tr>
</tbody>
</table>

2. To validate the installation select **Install Sametime Proxy Server** from the **Sametime Guided Activities** in the Integrated Solutions Console and the status of the existing deployment should be **Installed/Registered/Federated**.
Enable trust between Sametime Community server and the Sametime Proxy server

If the Sametime Community Server and the Sametime Proxy Server are using different IP addresses, as would be the case if they were on separate machines, then it would be necessary to configure the Sametime Community Server to trust the IP address of the Sametime Proxy Server. Refer to the following document for more information:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Managing_trusted_IP_addresses_sta852

In the example environment the two servers are located on the same machine and use the same IP address so this step is not necessary.

To validate that the Sametime Proxy server can access the Sametime Community server:
1. Enter the following URL into a browser:
2. Click on the button to Launch Sametime
3. Log in to Sametime with credentials of an LDAP user.

If a separate browser window opens to show the user as "Available", the Sametime proxy server is working.

Bind Ports to host name

As the server is hosting three different services that use three different host names, it is good practice to use the Integration Solutions Console to bind the appropriate ports to their host names to ensure that each service listens only on its own ports. This section details how to bind the ports used by the Sametime Proxy server to the Sametime Proxy server host name.
1. Create a Virtual Hosts configuration.
   a. Select Environment → Virtual hosts → New and give the configuration a name of STProxy_host and click OK.
   b. Click on the newly created virtual host link and select Host Aliases.
   c. Select New and enter the host name used in the Sametime proxy configuration and the associated port and click OK.
   d. Do the same to create host aliases for the port 80 and the associated https ports, 9444 and 443.
   e. Save the changes to the master configuration.

2. Configure Sametime Proxy application to use the STProxy_host definition.
   a. Select Applications → Application Types → WebSphere enterprise applications.
   b. Click on the link for the SametimeProxy application.
c. Click the **Virtual hosts** link in the Web Module Properties section.
d. In all the application modules, select **STProxy_host** as the virtual host and click **OK**.

e. Save the changes to the master configuration.

   a. Select **Servers → Server Types → WebSphere application servers** and click on the link for the **STProxyServer**.
   b. Click on the **Ports** link in the Communications section.
   c. Click on the **WC_defaulthost** link in the Port Name column.
   d. Enter **webchat.demos.ibm.com** as the Host. and click **OK**.

   e. Repeat the same step for the **WC_defaulthost_secure** entry.
44

f. Save the changes to the master configuration.

4. Restart the Sametime Proxy server.

**Sametime Meeting server**
The Sametime Meeting server hosts persistent meeting rooms which can be accessed directly or by starting a meeting from an instant messaging chat. This section details the steps to install the Sametime Meeting server on a separate server to the Sametime System Console, Sametime Community and Sametime Proxy servers.

**Create Meeting server database**
This section details the creation of the Meeting Server database.

1. First the database that will hold all the information on meeting rooms is created in DB2. For detailed information on this task please refer to the following document:  
   [http://www-10.lotus.com/ldd/stwiki.nsf/dx/Creating_a_database_for_the_Sametime_Meeting_Server_st852](http://www-10.lotus.com/ldd/stwiki.nsf/dx/Creating_a_database_for_the_Sametime_Meeting_Server_st852)

2. To validate that the database has been created, open the DB2 command center. If the STMS database is displayed, the installation is successful.

![DB2 command center screenshot](image)

**Create Meeting server deployment plan**
In this section the Sametime System Console is used to create a deployment plan for the Sametime Meeting server. For more detailed information please refer to the following document:

1. First it is necessary to connect the Sametime System Console to the Meeting Server database.
   a. From the Integrated Solutions Console select **Connect to DB2 Databases** from the **Sametime Prerequisites**.
   b. Complete the dialog using the following information.

<table>
<thead>
<tr>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostName</td>
<td>ssc.demos.ibm.com</td>
<td>Hostname of the DB2 server</td>
</tr>
<tr>
<td>Port</td>
<td>50000</td>
<td></td>
</tr>
<tr>
<td>Database name</td>
<td>STMS</td>
<td>Name of DB2 database created to hold the Meeting Server information</td>
</tr>
<tr>
<td>Application user ID</td>
<td>db2admin</td>
<td>Name and password of the DB2 administrator ID</td>
</tr>
<tr>
<td>Application password</td>
<td>password</td>
<td></td>
</tr>
</tbody>
</table>
2. To create the deployment plan, select **Install Sametime Meeting Server** from the **Sametime Guided Activities** in the Integrated Solutions Console.

   a. For the example environment, the default options were selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Sametime Meeting</td>
<td>Deployment Name</td>
<td>stmeeting</td>
<td></td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>Deployment Product Version</td>
<td>8.5.2</td>
<td></td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>Configuration</td>
<td>Primary node</td>
<td></td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>Deployment Manager</td>
<td>System</td>
<td>The admin ID and credentials for the Meeting server</td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>Host Name</td>
<td>meeting.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>User ID</td>
<td>wasadmin</td>
<td></td>
</tr>
<tr>
<td>WebSphere Profile Settings</td>
<td>Password</td>
<td>password</td>
<td></td>
</tr>
<tr>
<td>Choose Database</td>
<td>Database Name</td>
<td>STMS</td>
<td></td>
</tr>
</tbody>
</table>

**Install Meeting server**

In this section the Sametime Meeting server is installed.

1. For the example environment, the default install options are selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install Packages</td>
<td>Package Group</td>
<td>Use existing</td>
<td></td>
</tr>
<tr>
<td>Install Packages</td>
<td>Use System Console to install</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WAS Location</td>
<td>Use Sametime installed WebSphere Application Server</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSC Login</td>
<td>Host Name</td>
<td>ssc.demos.ibm.com</td>
<td>The host name for the Sametime System Console</td>
</tr>
<tr>
<td>SSC Login</td>
<td>Host Name of this computer</td>
<td>meetings.demos.ibm.com</td>
<td>The host name alias to be associated with the Sametime Meetings server</td>
</tr>
<tr>
<td>Deployment Plan List</td>
<td>Dep Name</td>
<td>stmeeting</td>
<td>The name of the deployment plan created in the Sametime System Console</td>
</tr>
</tbody>
</table>

2. To validate the installation select **Install Sametime Meeting Server** from the **Sametime Guided Activities** in the Integrated Solutions Console and the status of the existing deployment should be **Installed/Registered/Federated**.
3. To check that the Meeting server is operational, enter the following URL into a browser window...

http://meeting.demos.ibm.com:9080/stmeetings/

...and login using the credentials of one of the LDAP users. If the Welcome to Sametime Meeting Rooms screen is displayed, the Meeting server is operational.

**Bind Ports to hostname**

As with the Sametime Proxy server, it is good practice to use the Integration Solutions Console to bind the appropriate ports to their hostnames to ensure that the Meeting service listens only on its own ports. This section details how to bind the ports associated with the Sametime Meeting server to the Sametime Meeting server hostname.

1. Create a Virtual Hosts configuration
   a. Select Environment → Virtual hosts → New and give the configuration a name of STMeeting_host and click OK.
   b. Click on the newly created virtual host link and select Host Aliases.
   c. Select New and enter the host name used in the Sametime Meeting configuration (meeting.demos.ibm.com) and the associated port (9080) and click OK.
d. Do the same to create host aliases for the port 80 and the associated https ports, 9443 and 443.

<table>
<thead>
<tr>
<th>Virtual Hosts</th>
<th>STMeeting_host</th>
<th>Host Aliases</th>
<th>meeting.demos.ibm.com</th>
<th>9080</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>meeting.demos.ibm.com</td>
<td>9443</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>meeting.demos.ibm.com</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>meeting.demos.ibm.com</td>
<td>443</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>total: 4</td>
<td></td>
</tr>
</tbody>
</table>

e. Save the changes to the master configuration.

2. Configure Sametime Proxy application to use the STProxy_host definition.
   a. Select Applications → Application Types → WebSphere enterprise applications.
   b. Click on the link for the Sametime Meeting Server application.
   c. Click the Virtual hosts link in the Web Module Properties section.
   d. In all the application modules, select STMeeting_host as the virtual host and click OK.
e. Save the changes to the master configuration.

   a. Select Servers → Server Types → WebSphere application servers and click on the link for the STMeetingServer.
   b. Click on the Ports link in the Communications section.
   c. Click on the WC_defaulthost link in the Port Name column.
   d. Enter meeting.demos.ibm.com as the Host. and click OK
e. Repeat the same step for the **WC_defaulthost_secure** entry.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>WC_defaulthost</td>
<td>meeting.demos.ibm.com</td>
<td>9080</td>
</tr>
<tr>
<td>WC_defaulthost_secure</td>
<td>meeting.demos.ibm.com</td>
<td>9443</td>
</tr>
<tr>
<td>Total</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>

f. Save the changes to the master configuration.

4. Restart the Sametime Meeting server.

**Configure the Sametime Meeting server to use the Sametime Proxy server for presence awareness and chat**

In order to use live names in Sametime meetings for presence awareness and instant messaging there are some additional pieces of configuration to complete. These are:

- Enable trust between the Sametime Community server and the Sametime Meeting server.
- Setup single sign-on between the Sametime Meeting server and the Sametime Community server.
- Define a Sametime Proxy server for awareness in meeting rooms.

**Enable trust between Sametime Community server and the Sametime Meeting server**

In order for the Sametime Meeting server to be able to use the Sametime Community server for presence awareness and chat it is necessary to configure the Sametime Community Server to trust the IP address of the Sametime Meeting Server. Refer to the following document for more information:


**Setup single sign-on**

In the example environment single sign-on is required across the whole environment and is configured after all the individual servers have been installed. This is described in Chapter 5, “Implement and test single sign-on” on page 95.

**Define the Sametime Proxy server for the Sametime Meeting server**

The Sametime Proxy server to be used by a Sametime Meeting server is defined in the Sametime System console. Refer to the following document for more information:


1. In the example environment, in Sametime System Console > Sametime Servers > stmeeting, the configuration key meetingroomcenter.stProxyAddress is set to http://webchat.demos.ibm.com:9081

**Define the Sametime Meeting server for the Sametime Proxy server**

The Sametime Meeting server to be used by a Sametime Proxy server is defined in the Sametime System console. Refer to the following document for more information:
1. In the example environment, in **Sametime System Console → Sametime Servers → Sametime Proxy Servers → 852proxy**, the Sametime Meeting server is configured as follows:

<table>
<thead>
<tr>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of Meeting server</td>
<td>Sametime Meeting Server</td>
<td></td>
</tr>
<tr>
<td>Host Name</td>
<td>meeting.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td>Port</td>
<td>9080</td>
<td></td>
</tr>
</tbody>
</table>

**Validating the interoperability between the Sametime Proxy server and the Sametime Meeting server**

Before testing the configuration it is advisable to restart the Sametime Meeting and Sametime Proxy servers.

In order to verify that the Sametime Meeting is able to use Live names for presence awareness and chat:

1. Login to the Meeting Room server at [http://meeting.demos.ibm.com:9080/stmeetings](http://meeting.demos.ibm.com:9080/stmeetings) using the credentials of one of the LDAP users.

2. Create a meeting room for the user.

If the presence awareness icon is displayed against the user's name in the list of meeting rooms, the Sametime Meeting server is using presence awareness.

Presence awareness and instant messaging features are also available from the Participants panel.
In order to verify that the Sametime Proxy server can use the Sametime Meeting server for instant meetings:

1. Login to the Sametime web client with the credentials of one of the LDAP users who is the owner of at least one meeting room defined on the Sametime Meeting server.
2. Login to the Sametime web client from another browser with a different set of credentials.
3. From the first client, add the user of the second client to the contact list (if they are not already added)
4. Right-click on the second user's name and select Invite to Meeting Room.

The functionality is working for the use of existing meeting rooms if:

1. The user can select to use an existing meeting room and is offered the list of meeting rooms for which they are the owner.
2. When the user clicks Invite, an instant message is sent to the user with a link to the selected meeting room.

3. Both users arrive in the designated meeting room.

4. The meeting room and any contents are retained when the meeting is finished.

The functionality is working for the use of instant meeting rooms if:

1. The user can select to use an instant meeting room.

2. When the user clicks invite, an instant message is sent to the user with a link to the instant (temporary) meeting room.

3. Both users arrive in the meeting room

4. The meeting room and any contents are not retained when the meeting is finished.
4.3 IBM Connections installation

4.3.1 Overview

IBM Connections is social software for business that helps you to find and connect with people and information in your organization.

IBM Connections capabilities are delivered through a set of integrated services.

- **Profiles**: Add your skills, experience and contact details. Find and connect with the people who have the expertise or experience you need.
- **Files**: Upload, tag and share your files. Use tags to find, download and re-use other people's files.
- **Bookmarks**: Save, tag and share your bookmarks. Use tags to search for and find bookmarks others have posted.
- **Blogs**: Post your ideas and experiences in a weblog and get feedback from others.
- **Communities**: Create, find, join, and work with communities of people with whom you have a shared interest.
- **Wikis**: Collaboratively edit content with a team on a specific topic.
- **Forums**: Create a discussion area where people can ask questions, share their experiences, and discuss topics of common interest.
- **Activities**: Set up and manage team tasks.
- **Home page**: One place to go for updates from all the IBM Connections services.

Server diagram

The example environment built for this project consists of a single-server Connections deployment. In a production environment, it is very likely that the services described below would be spread over a number of servers and duplicated for performance, scalability and high availability purposes.

The Connections server has a DB2 data store which, along with the file system, is used by Connections to store the data associated with each of its components.

Tivoli Directory Integrator is used to populate the DB2 database which stores profile information (PEOPLEDB) with information from the LDAP. In the example environment, this is used to create a set of Connections users with a management hierarchy, job roles and responsibilities, and location information. This provides a set of realistic characters for the purposes of demonstrating real-life scenarios with the integrated environment. In a production environment, it is likely that Tivoli Directory Integrator would be configured to update the profile information at regular intervals so that changes in the organization's directory would be automatically reflected in the Connections profiles. However, in the example environment, the Tivoli Directory Integrator does not have an ongoing active role.

The Connections server itself is deployed on WebSphere Application Server in a deployment manager and cell configuration, with a node agent to facilitate communication and synchronization between the deployment manager and the other servers in the cell. The server which supports the Connections applications is created as part of a cluster. In a production environment, the cluster would be likely to contain more than one server, on more than one machine in order to offer load balancing and failover for the Connections services. In the example environment, there is a single server in this cluster. The default "server1" application server is useful during the installation to verify the deployment but does not have
an active role in the day-to-day operation of Connections and is not running on the system in the demonstration scenarios.

An HTTP server is installed alongside the WebSphere Application Server and allows clients to access Connections over the standard HTTP/HTTPS ports (80/443) and an LDAP hosted on the iNotes (Domino) server is used for authentication.

The dotted lines in the diagram indicate elements that were used during the configuration and testing of the Connections server but are not part of the ongoing operation.

The unbroken lines in the diagram indicate the communication paths used in operating the system.

**Prerequisites**

For information on the basic hardware and operating system configuration of the server, refer to Chapter 3, “Prerequisite information” on page 21.

For more information on specific software and hardware requirements for Connections, please refer to the following technote:

http://www-01.ibm.com/support/docview.wss?rs=899&uid=swg27012786

IBM Connections 3.01 is installed in the example environment. The following is the complete list of products and part numbers required for this specific installation (for both IBM Connections and the prerequisite software).
4.3.2 Prerequisite installation and configuration

This section describes, at a high level, the steps required to install the prerequisite software for the IBM Connections server with any non-default choices that were made. Where appropriate, links to more detailed information on these steps are provided.

As described in Chapter 3, “Prerequisite information” on page 21, the default directory names are not used.

**DB2 server**

Connections stores information on each of its components in a relational database. In the example environment the database used is DB2.

This section gives an overview of the high level steps involved to install DB2 V9.7 Fixpack 2.

For more detailed information please refer to the following document:


Note that the fix pack for DB2 contains all the DB2 server code so it is not necessary to install DB2 9.7 and then install the fix pack over the top. However, the fix pack comes with a short-term trial license installed and this will need to be updated this with a permanent licence as DB2 will stop working when the trial license expires.

This document describes how to apply the DB2 license:
For the example environment, the default install options were selected with the exception of the following:

- Notifications are disabled (In a production environment notifications are likely to be sent to a specific administrator account).

To validate the installation, check the system tray. If DB2 has been installed successfully it will be started and the DB2 Control Center icon will appear in the system tray.

**Tivoli Directory Integrator**

Tivoli Directory Integrator (TDI) is used to populate the Connections profiles from one or more data sources in the organization. For the example environment, TDI was used to populate the profiles from information held in the Domino LDAP.

This section gives an overview of the high level steps involved to install Tivoli Directory Integrator V7.0 with fix pack 5.

For more detailed information please refer to the following document:


For the example environment, the default options were selected with the exception of the following:

- Custom installation: Features to install: **Runtime Server** and **Configuration Editor** only (Not all the features are required for a one-time upload of entries from the LDAP directory).

When the installation is complete, apply the fix pack from the TDIInstallDir\bin directory using the command:

```
applyUpdates -update path and filename of fix
```

To validate the installation type:

```
applyUpdates -queryreg
```

The resulting output indicates the version of the installed components.
WebSphere Application Server

Connections is deployed as a server on WebSphere Application Server.

This section gives a high level overview of the steps required to install WebSphere Application server as a prerequisite for Connections.

For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Installing_IBM_WebSphere_Application_Server_ic301

For the example environment, the default options were selected with the exception of the following:

- WebSphere Application Server environment: Cell (deployment manager and a managed node) (This is the required configuration for Connections)

At the end of the installation the completion of the install can be verified from the First Steps page.
To validate the operation of the installation, open a browser and enter the URL for the snoop servlet. The snoop servlet is a small application that is installed by default with WebSphere Application Server and is a useful tool for validating installation and configuration.

http://host name:9080/snoop

If this URL resolves correctly, then the WebSphere Application Server is operational.

Snoop Servlet - Request/Client Information

Requested URL:

http://connections.demos.ibm.com:9080/snoop

In order to test the Deployment manager, enter the following URL for the Integrated Solutions Console:

http://host name:9060/admin

If the login screen displays (after accepting the SLL certificate) then the Deployment manager is operational.

IBM HTTP server

Installing an HTTP Server in the Connections environment allows users to access the Connections server without having to add port numbers to the URLs of the Connections components.

This section gives an overview of the high level steps to install IBM HTTP Server.

For more detailed information please refer to the following document:


The HTTP server (and plugin) can be installed from the WebSphere Application Server installation screen or directly from the install code directory.

For the example environment, all the default options were selected.

To validate the installation, open a browser and enter the following URL.
http://hostname

If the Welcome page for the HTTP server is displayed, the HTTP server is operational.

Apply fix pack

WebSphere Application Server and IBM HTTP server usually need a fix pack or other fixes applied to support specific applications. Refer to the detailed system requirements for confirmation of the fix pack and/or fixes required for Connections:

http://www-01.ibm.com/support/docview.wss?rs=899&uid=swg27012786

This section gives an overview of the high level steps to apply the fix pack.

For more detailed information please refer to the following document:


**Before applying a fix pack:**

1. Stop the WebSphere Application Server.
2. Install the latest version of the WebSphere Application Server Update Installer.
   
   For the example environment, all the default options were selected.
3. Copy the WebSphere Application Server fixes into the `InstallDir\UpdateInstaller\maintenance` directory. For the example environment these were:
   
   - 7.0.0-WS-IHS-WinX64-FP0000017
   - 7.0.0-WS-PLG-WinX64-FP0000017
   - 7.0.0-WS-WASSDK-WinX64-FP0000017
   - 7.0.0-WS-WAS-WinX64-FP0000017
To install the fix pack:
Run the Update Installer three times to apply all the applicable fixes to each of the following product locations in turn. Note that some of the above fixes will be applied to more than one of the product locations. Each time, all the fixes that can be selected should be selected (some may be greyed out).

1. E:\IBM\WebSphere\AppServer
2. E:\IBM\HTTPServer
3. E:\IBM\HTTPServer\Plugins

After applying the fix pack:
1. Restart the HTTP server.
2. Restart the WebSphere Application Server.

To validate the installation open the Integrated Solutions Console from the browser...
http://host name:9060/admin

...and check the About this Integrated Solutions Console panel on the right of the screen:

If the correct fixpack level is shown, the server has been successfully upgraded.

Configure the federated repository
Administrative security was enabled as part of the WebSphere Application Server installation. This means that a federated repository has been configured with one user (the administrator ID that was provided during the installation). In this section, the Domino LDAP directory will be added to the federated repository, and application security will be enabled.

For more detailed information on adding a federated repository please refer to the following document:

http://www-10.lotus.com/ldd/1cwiki.nsf/dx/Setting_up_federated_repositories_ic301

1. Verify LDAP connectivity to the Domino LDAP from the Connections server. This can be done using LDAP commands or an LDAP administrator tool.
2. From the Integrated Solutions Console (Security → Global security):
   a. Enable application security and click Apply and then the save link.
   b. Configure a federated repository. For the example environment, the following values were used. Then click Apply and the save link.
c. Add the base entry details. For the example environment, the following values were used. Then click **Apply** and the **save** link.

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LDAP server</strong></td>
<td>Repository identifier</td>
<td>dominoldap</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Directory type</td>
<td>IBM Lotus Domino</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Primary host name</td>
<td>domino.demos.ibm.com</td>
<td>Host name of the Domino server.</td>
</tr>
<tr>
<td><strong>Security</strong></td>
<td>Bind distinguished name</td>
<td>cn=wpcbind</td>
<td>This is not required if the LDAP allows anonymous access.</td>
</tr>
<tr>
<td></td>
<td>Bind password</td>
<td>password</td>
<td>Password configured as internet password in person document for the Bind ID.</td>
</tr>
<tr>
<td></td>
<td>Login properties</td>
<td>uid;cn</td>
<td>uid must be the first entry but adding cn allows users to login with their common name. The entry mail can also be added if users are to be able to login with their email address. Any entries should be separated by a semi-colon.</td>
</tr>
</tbody>
</table>

d. Restart the Deployment manager and the application server.

To validate the changes, open a browser window and access the snoop application again:

http://hostname:9080/snoop

This time a user name and password must be provided before the site can be accessed.

**Configure HTTP Server and plugin**

The IBM HTTP Server serves HTML pages from the file system. The WebSphere plugin passes URLs through from port 80 to the WebSphere Application Server on other ports e.g.
9080. This section details the high level steps to define the HTTP Server in WebSphere Application Server and configure the plugin.

For more detailed information on configuring the HTTP Server and plugin please refer to the following document:


1. From the Integrated Solutions Console (Servers → Server Types → Web Servers), create a new server. For the example environment, the following values were used. Then click Finish and the save link.

<table>
<thead>
<tr>
<th>Step</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Select node</td>
<td>ConnectionsNode01</td>
<td>The node created when the WebSphere Application Server was installed (the name is automatically derived from the short name of the server appended with &quot;Node01&quot;).</td>
</tr>
<tr>
<td></td>
<td>Server name</td>
<td>webserver1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type</td>
<td>IBM HTTP Server</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>Web server template</td>
<td>IHS</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td></td>
<td></td>
<td>Accept the defaults.</td>
</tr>
</tbody>
</table>

The definition is complete, and the Plugin Configuration XML file should be automatically copied to the HTTP server.

2. Restart the HTTP server.

To validate the configuration, open a browser window and access the snoop application again, but this time do not specify a port (port 80 will be used by default)

http://host name/snoop

The configuration completed successfully if a user name and password are required and then, when these have been validated, the snoop application is displayed.

If there is a problem with the configuration, try manually generating and propagating the plugins from the Integrated Solutions Console (Servers → Server Types → Web servers).

**Configure HTTP Server for SSL**

Connections requires that SSL be configured so that passwords are encrypted when logging in. By default, Connections will switch to HTTPS for the login screen, and then switch back to HTTP.

This section gives the high level steps to configure the HTTP Server with an SSL certificate.
For more detailed information on this topic, please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Configuring_IBM_HTTP_Server_for_SSL_ic301

1. Using the `ikeyman` utility (InstallDir/bin directory), create a key database and a self-signed certificate. For the example environment, the following values were used.

<table>
<thead>
<tr>
<th>Step</th>
<th>Entry</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>New key database file</td>
<td>Key database type</td>
<td>CMS</td>
</tr>
<tr>
<td></td>
<td>Filename</td>
<td>key.kdb</td>
</tr>
<tr>
<td></td>
<td>Location</td>
<td>E:/IBM/HTTPServer/conf</td>
</tr>
<tr>
<td>Password prompt</td>
<td>Password</td>
<td>password</td>
</tr>
<tr>
<td></td>
<td>Slash password to a file</td>
<td>Yes</td>
</tr>
<tr>
<td>New self-signed certificate</td>
<td>Key Label</td>
<td>LC301 Demo</td>
</tr>
</tbody>
</table>

2. Edit the HTTP Server Configuration file to enable SSL and to use the new certificate database.
   a. Copy and paste the SSL enablement section.
   b. Remove the comments from around the pasted section and edit it to match the configuration.

   For the example environment, the section looks like this:

   ```
   LoadModule ibm_ssl_module modules/mod_ibm_ssl.so
   Listen 0.0.0.0:443
   <VirtualHost *:443>
       ServerName connections.demos.ibm.com
       SSLProtocol Disable SSLv2
   </VirtualHost>
   KeyFile E:/IBM/HTTPServer/conf/key.kdb
   SSLDisable
   # Enable IBM HTTP Server diagnostic features.
   ```

   c. Stop and restart the IBM HTTP Server.

To validate the configuration, open a browser, access the following URL (using HTTPS rather than HTTP) and accept the certificate

https://host name

If the configuration completed successfully there should be a padlock icon at the bottom of the browser screen. Click on the padlock icon to see details of the site security.
4.3.3 Deploy Connections

This section gives the high level steps to deploy IBM Connections once the prerequisite environment has been installed and configured.

Installation
This section gives the high level steps to install Connections including creating the databases, installing the code and applying fixes.

Create databases
Connections stores information on each of its components in a relational database. In the example environment the database used was DB2.

This section gives the high level steps to create the databases.

For more detailed information on this topic, please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Creating_databases_with_the_database_wizard_ic301

1. Create a user, LCUSER as a local operating system account. (LCUSER is the user name with which the wizard will create the databases).
2. Add the user to the DBUSERS group. (This is the group that have access to databases in DB2).
3. Run the dbwizard30.bat Connections wizard. For the example environment, the following values were used.
To validate the installation, open the DB2 Command Center and look for the databases. If there are nine databases, as shown in the screenshot below, then the database creation was successful.

![DB2 Command Center screenshot](image)

**Populate profiles**

Once the databases are created, the profiles database (PEOPLEDB) can be populated with people information from other sources in the environment. In the example environment the data source is the Domino LDAP.

In the example environment's Domino LDAP, person records hold additional information about each character's working environment, including job titles, manager and assistant details, phone numbers and location information.
This information can be reflected in the Connections profiles by using Tivoli Directory Integrator to map these fields with fields in the Connections profile.

This section gives the high level steps to import the people information from the Domino LDAP into Connections profiles. For more detailed information on this topic, please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Adding_LDAP_data_to_the_Profiles_database_ic301

1. Run the Connections wizard populationWizard. For the example environment the default values were used with the following additional entries on the Profiles database mapping. Some of the attributes are from the `inetOrgPerson` schema and some are from the `dominoPerson` schema as indicated in the table below.

<table>
<thead>
<tr>
<th>Database field</th>
<th>LDAP Schema</th>
<th>LDAP attribute</th>
<th>Person document field (as displayed in screen above)</th>
</tr>
</thead>
<tbody>
<tr>
<td>deptNumber</td>
<td>dominoPerson</td>
<td>Department</td>
<td>Department</td>
</tr>
<tr>
<td>employeeNumber</td>
<td>dominoPerson</td>
<td>Employee ID</td>
<td>Employee ID</td>
</tr>
<tr>
<td>jobResp</td>
<td>inetOrgPerson</td>
<td>Title</td>
<td>Title</td>
</tr>
<tr>
<td>officeName</td>
<td>dominoPerson</td>
<td>Location</td>
<td>Location</td>
</tr>
<tr>
<td>orgld</td>
<td>dominoPerson</td>
<td>Company Name</td>
<td>Company</td>
</tr>
<tr>
<td>pagerNumber</td>
<td>inetOrgPerson</td>
<td>pager</td>
<td>Pager number</td>
</tr>
<tr>
<td>workLocationCode</td>
<td>inetOrgPerson</td>
<td>physicalDeliveryOfficeName</td>
<td>Office number</td>
</tr>
</tbody>
</table>

On the optional database tasks screen the following were selected for the example environment:

- Countries (using the default .CSV file).
Work locations (using a custom CSV file that maps the Office number field in Domino to a business address as shown in the screenshot below).

Mark profiles of each manager (required for manager hierarchy links in profile).

To validate the import, open the DB2 command center and examine the EMPLOYEE table in the PEOPLEDB database. If the import completed successfully, there should be a record in this table for each person in the LDAP.

The example environment contains a directory of photos for each of the Connections users...

In addition to importing person information into Connections profiles, it is possible to import a photo for each profile record.

The next part of this section gives the high level information on how to import photos into Connections profiles. For more detailed information on this topic, please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Populating_Profiles_with_photos_ic301

The example environment contains a directory of photos for each of the Connections users...
...and a collect_photos.in file which contains details of where each photo is stored along with the uid of the user to whom the photo belongs.

The batch file included with the Connections population wizards (load_photos_from_files) was used to add each of the photos to its respective owner's profile.

Although it is not possible to view the photos in DB2, it is possible to validate the import by examining the content of the PHOTO table in the PEOPLEDB. Here there should be an entry for each user. The entry is linked to the record in the EMPLOYEE table using the PROF_KEY field.
Install Connections server code

Once the databases are created, the Connections server can be installed.

This section gives the high level steps to install the Lotus Connections server code.

For more detailed information on this topic, please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Installing_IBM_Connections_3.0.1_ic301

Install the Connections server using the Connections wizard. For the example environment, the following values were used.
To validate the installation, open a browser window and type in the following URL.

http://host name/homepage

If the home page of the Connections server is displayed, the installation was successful. Login to Connections using the credentials for one of the LDAP entries and any information that was imported into the profile using Tivoli Directory Integrator should be displayed, including photo, location and contact information, and management hierarchy.
Apply Fixes

Connections may need fixes applied in order to resolve issues found after the product was released. Refer to the Fix Central for details of any fixes required.

http://www-933.ibm.com/support/fixcentral/

This section gives an overview of the high level steps to apply the interim fixes required for Connections 3.0.1

For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Installing_interim_fixes_ic301

The following are the interim fixes that were applied to the example environment:

- 3.0.1.0-IC-Multi-UPDI-3.0.1.0_20110506_1258 (IBM Connections 3.0.1.0 Update Installer)
- 3.0.1.0-LC-Multi-SEARCH-CRLO61763 (Fix for search)
- 3.0.1.0-IC-Multi-Comm-IFLO62783 (Fix for communities)
- 3.0.1.0-IC-Multi-Comm-IFLO62035 (Add support for Quickr Library Widget to Communities)
- 3.0.1.0-IC-Multi-Comm-IFLO62399 (Fix for communities)
- LO62605 (Fix for Sametime awareness in profiles - https://www-304.ibm.com/support/docview.wss?crawler=1&uid=swg1LO62605)
Before applying the fixes:
1. Stop the Connections server.
2. If required, install the latest version of the Connections Update Installer. (For Connections 3.0.1 this is supplied as the first interim fix in the list above). For the example environment, all the default options were selected.
3. Copy the other Connections interim fixes into the installDir\UpdateInstaller\fixes directory.

To apply the fixes:
1. Run updatewizard.bat from the installDir\updateInstaller directory. For the example environment, all of the default options were selected.

After applying the fixes:
1. Synchronize the node using the Integrated Solutions Console (System Administration → Nodes)
2. Restart the Connections server.

Configuration
Once Connections has been installed, there is some additional configuration required. For the example environment the following post-installation tasks were completed.

For more detailed information on all post-installation tasks please refer to the following document:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Mandatory_postinstallation_tasks_ic301

1. Configuring Connections to use HTTPS. This involves:
   a. Mapping the Connections application modules to the HTTP Server. See the following document for details:
      http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Mapping_applications_to_IBM_HTTP_Server_ic301
   b. Removing the port numbers from the web addresses in HTTP server. See the following document for details:
      http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Updating_web_addresses_in_IBM_HTTP_Server_ic301
   c. Adding the self-signed certificate to the trust store. See the following document for details:
      http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Adding_certificates_to_the_WebSphere_trust_store_ic301
2. Configuring an administrative user for the Home Page and Blogs. See the following document for details:
   http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Configuring_an_administrative_user_ic301
   For the example environment the Blogs administrator is Betty Zechman
3. Creating a home page for Blogs. See the following document for details:
For the example environment the home page for Blogs is
http://connections.demos.ibm.com/blogs/home

4.4 Websphere Portal installation

This section describes installation of the Portal server to the level used in the integration environment.

4.4.1 Overview

WebSphere Portal provides core portal services that aggregate applications and content and deliver them as role-based applications. The features and functionality include:

- Flexible access to existing enterprise data and applications, as well as to external news feeds or Web data
- Enhanced tagging and rating allowing users to easily organize and evaluate content without leaving the page they are viewing
- REST services that allow you to incorporate composite mashup applications, including service feeds from other Web applications
- A site wizard that allows users to create and deploy independent virtual portals
- Templates, a Theme Customizer portlet and WebSphere Portlet Factory software that help you speed application deployment through new portlet development and deployment capabilities
- Integration and tooling that allow you to quickly create composite, high-value applications that leverage existing investments in legacy applications and databases
- A Page Builder that simplifies page creation, making it easy for business users to create pages and add content to pages

There are three different Portal offerings:

- Server
- Enable
- Extend

Any of the Portal offerings could be used to integrate with the other products in this demo. The biggest difference between the offerings is the Server offering does not include WCM and Enable/Extend does include WCM. The Portal Extend offering is installed in the example environment because it is the most comprehensive offering with respect to features.

For more detailed information on the different Portal server offerings, see the following documentation:
Server diagram
The Portal server used in this demo is configured with DB2 and IBM HTTP server. Here is a diagram of this server:

Prerequisites
For information on the basic hardware and operating system configuration of the server, refer to 3. Prerequisite information.

For more information on software and hardware, refer to the following technote:


Downloading the install image
Download the following part numbers to install Portal, IBM HTTP Server and DB2:
For more information on downloading the WebSphere Portal install image, refer to the following technote:


4.4.2 Installation

After you have downloaded the installation images, you need to extract them to a directory, ie. install_image.

Portal Server

To begin the installation, run install_image/install.bat. During the installation, the default options were selected with the exception of the following:

Product to install: IBM WebSphere Portal Extend

Installation type: Full install, Note: A migration from v6.1 to v7 requires a full install of the v7 server.

Installation Directory: F:\WebSphere

Node name: node1

Fully qualified host name: portal.demos.ibm.com

Application server and portal administrative ID: wpsadmin

Application server and dportal password: password

DB2 Server

To begin the installation, run install_image/setup.exe. During the installation, the default options were selected with the exception of the following:

Installation Directory: F:\SQLLIB
User: db2admin
Password: password
Disable notifications

**IBM HTTP Server**
To begin the installation, run install_image/IHS/install.exe. During the installation, the default options were selected with the exception of the following:

Installation Directory: F:\HTTPServer
Windows Service Definition User: Administrator
Password: password
UserID for IBM HTTP Server administration server authentication: ihsadmin
Password: password

### 4.4.3 Configuration

After you have completed the installations described in the previous section, there are numerous configuration steps required in order to use Portal as a standalone server.

**Database transfer**
Portal is configured to use a local derby database after installation. This demo uses a DB2 database and the following documentation was used to configure DB2:


For reference, refer to the wkplc.properties attached to this article as a reference.

**Web server plugin**
As part of the IBM HTTP Server install, it will install the web server plugin and should create the webserver definition in WAS. A common problem is that the plugin install won't create the webserver definition in WebSphere Application Server. If this occurs, create the webserver definition manually using the WAS console. To create the webserver definition manually, do the following:

1. Log into the WAS administrative console. For more information, see section "Accessing Portal and the WebSphere administrative console" below.
2. Navigate to **Servers → Web servers**
3. Click **New**
4. Specify values for Step 1 as shown below:
Enable security

When you enable security on the Portal server, you can choose between a federated or standalone repository. If you want to configure multiple LDAP servers or create virtual portals, you should configure federated. This demo was configured to use a standalone repository with a Domino LDAP server. The following documentation was used to configure Domino LDAP:

http://www-10.lotus.com/ldd/portalwiki.nsf/dx/Windows_standalone_Configuring_WebSphere_Portal_to_use_a_user_registry_wp7
Lotus Domino uses a flat naming convention for its groups. Therefore it is important to ensure that the value specified for the baseDN (i.e. standalone.ldap.baseDN for standalone LDAP or federated.ldap.baseDN for federated LDAP) when configuring security is left blank, so that it will be inclusive of the Portal administrative group. For more information, refer to the following technote:


For reference, refer to the `wkplc.properties` attached to this article as a reference.

### 4.4.4 Installing fixpacks and fixes

The following section details the post-install tasks required to apply the relevant fix packs.

**Upgrade to WAS 7.0.0.13**

WAS 7.0.0.13 is a prerequisite to upgrading to Portal v7.0.0.1.

Download and install IBM Update Installer V7.0.0.19 for WebSphere Software for Windows. Here is a link:

http://www-01.ibm.com/support/docview.wss?rs=180&uid=swg24020448

Download and install 7.0.0.13: WebSphere Application Server V7.0 Fix Pack 13 for Windows. Here is a link:


**Upgrade to Portal 7.0.0.1**

Download and install IBM WebSphere Portal Update Installer. Here is a link:


Download and install WebSphere Portal and WCM V7.0 fix pack 1. Here is a link:


For more information, refer to this technote:


**Upgrade to the latest cumulative fix**

Download and install the latest cumulative fix. The server used in the example environment has CF07 installed. To install this combined cumulative fix, refer to the following instructions:


**Note:**

Remember to stop the HTTP server prior to installing any fixpacks or fixes.

Remember to generate and propagate the web server plugin file, and restart the HTTP server...
4.4.5 Verifying the install

After you have completed all the steps described in the previous sections, you should verify the installation.

Starting and stopping the server

To start the server, follow these steps:

1. Open a command prompt
2. Navigate to the wp_profile/bin directory.
3. Type `startServer.bat WebSphere_Portal` at the command prompt

You should get the following message:

```
F:\WebSphere\wp_profile\bin>startServer.bat WebSphere_Portal
ADM00101: Tool information is being logged in file
F:\WebSphere\wp_profile\logs\WebSphere_Portal\startServer.log
ADM01201: Starting tool with the wp_profile profile
ADM01500: Reading configuration for server: WebSphere_Portal
ADM03201: Server launched. Waiting for initialization status.
ADM03801: Server WebSphere_Portal open for e-business; process id is 1140
```

Similarly, to stop the server, follow these steps:

1. Open a command prompt
2. Navigate to the `wp_profile/bin` directory. For example, F:\WebSphere\wp_profile\bin
3. Type `stopServer.bat WebSphere_Portal -username wpsadmin -password password` at the command prompt

You should get the following message:

```
F:\WebSphere\wp_profile\bin>stopServer.bat WebSphere_Portal -username wpsadmin -password password
ADM00101: Tool information is being logged in file
F:\WebSphere\wp_profile\logs\WebSphere_Portal\stopServer.log
ADM01201: Starting tool with the wp_profile profile
ADM01500: Reading configuration for server: WebSphere_Portal
ADM03201: Server stop request issued. Waiting for stop status.
ADM04001: Server WebSphere_Portal stop completed.
```

Accessing Portal and the WebSphere administrative console

Once you have verified that the server has started successfully, you can log into Portal using the following URL:

http://portal.demos.ibm.com/wps/portal

Verify that you can log into Portal using the administrative account. In this demo, we used the following administrative account:

username: wpsadmin
password: password

To log into the WebSphere administrative console, access the following URL:

https://portal.demos.ibm.com:10032/admin
You should use the same administrative account to log into the WebSphere administrative console.

**Accessing the log files**

The main log files are located in the `wp_profile\logs\WebSphere_Portal` directory. All log statements are written to `SystemOut.log` and `SystemErr.log`. After the server starts successfully, you will see the following in `SystemOut.log`:

```
```

## 4.5 Lotus Quickr for Domino installation

### 4.5.1 Overview

IBM Lotus Quickr Domino is web-based team collaboration software that helps you share content and collaborate with teams.

Lotus Quickr Domino provides the following benefits:

- Help teams and workgroups become more efficient and more productive, including teams that span multiple geographies and time zones
- Share, access and collaborate on team content that is the most up-to-date.
- Empower teams to set up and manage their information and projects in a security-rich environment without requiring IT assistance.
- Provide an effective alternative to e-mail for storing and managing content — cut down on e-mail overload
- Reduce "inbox bloat" by providing a central, shared way to share attachments and large files
- Capture and reuse business best practices so that teams and projects can get up and running more quickly
- Provide an intuitive and easy-to-use front end to enterprise content management systems to get more out of them

Quickr Domino has a rich set of features, including:

- Team Places which provides places for people, team, projects,
- Content libraries to share information, team discussion forums to encourage communications
- Connectors that help make sharing easier, Replacing email attachment with shared links which connect team collaboration with other software.
- Lotus Quickr software can be Integrated with Lotus Notes, Lotus iNotes, IBM Connections, IBM Sametime, Lotus Connections, Lotus Symphony, WebSphere Portal & other third party products
- Support for Domino Off-Line Services, which enables users to create a copy of a Quickr Place on their local machine, make changes to the local copy without a network connection, and then synchronize changes with the server when reconnected to the network (Microsoft Windows edition only).
Please refer to the following document to understand more on what's new in Lotus Quickr 8.5.1, http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Whats_new_in_this_release_qd851

Server diagram
The example environment built for this project consists of a single-server Quickr Domino deployment. In a large-scale production environment, it is very likely that the services described below would be spread over a number of servers and duplicated for performance, scalability and high availability purposes.

Prerequisites
For more information on software and hardware requirements, refer to the following technote: https://www-304.ibm.com/support/docview.wss?rs=3264&uid=swg27020463

IBM Lotus Quickr for Domino 8.5.1 was installed in the example environment. The following is the complete list of products and part numbers required for this specific installation (for both Quickr Domino and the prerequisite software).

Note: Quickr Domino 8.5.1 only supports the 32 bit Lotus Domino Server
4.5.2 Installation

This section gives an overview of the high level steps involved to install Domino 8.5.1 and Quickr Domino 8.5.1 Server.

Notice: Quickr Domino servers require a dedicated Domino server. If you use a Domino server for anything, such as mail, then configure a new Domino server for use by Quickr Domino.

Install Domino 8.5.1 Server
This section describes the installation of the Domino server that will host the Quickr Domino service.

1. Before starting the Domino Installation for the Quickr Domino service, verify that you can connect to the primary Domino server from which the system databases will be pulled.
2. From a command prompt type:
   
telnet host name of the primary Domino server 1352

   ![Command Prompt]

   C:\>telnet domino.demos.ibm.com 1352

   The result should be a blank Telnet window

3. The Domino server is installed using all the default options. For more detailed information please refer to the following document:
   http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Quickr_Domino_8.5_server_installation_and_configuration

4. In the Domino setup the following non-default choices are made:
Install Domino 8.5.1 fix pack 5
This section gives an overview of the high level steps involved to install Domino 8.5.1 Fix pack 5. For more detailed information please refer to the following document:

http://www-01.ibm.com/support/docview.wss?rs=899&uid=swg24025721#851FP5

1. The Domino 8.5.1 fix pack is installed using all the default options.

Verify Domino server installation
Start the Domino Server to verify the installation. The server should start and show the correct version number.

Install Quickr Domino 8.5.1 server
This section gives an overview of the high level steps involved to install Quickr Domino 8.5.1 with Fixpack 8

Note: Stop the Domino server (on which the Quickr Domino service is to be Installed) before starting the installation

For more detailed information please refer to the following document:

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Quickr_Domino_8.5_server_installation_and_configuration on Part 3

For the example environment, the default options are selected with the exception of the following:

<table>
<thead>
<tr>
<th>Panel</th>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installation</td>
<td>Lotus Quickr hostname</td>
<td>quickr.demos.ibm.com</td>
<td>The id should be an ID that is NOT in the LDAP. This allows Quickr to be administered even if the LDAP server is not available</td>
</tr>
<tr>
<td></td>
<td>Lotus Quickr Administrator</td>
<td>quickadmin</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fully qualified hostname for this Quickr server</td>
<td>quickr.demos.ibm.com</td>
<td></td>
</tr>
</tbody>
</table>
Install Quickr Domino 8.5.1 fix pack 8
This section gives an overview of the high level steps involved to install Quickr Domino Fix Pack 8. Fix pack Installation on Windows requires the system path variable to be set and Java to be installed.

For more detailed information please refer to the following general document on installing fix packs for Quickr Domino on Windows:

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Configuring_Windows_Server_for_Quickr-D_FixPack_Installer

The Quickr Domino 8.5.1 fix pack is installed using all the default options.

Enabling Java Servlet engine
After you have installed Quickr Domino, you must enable the Domino Servlet engine. The Servlet engine must be enabled in order to use Quickr Domino place administration and to ensure that the Quickr Domino home page displays correctly.

For more detailed information please refer to the following document:
http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Enabling_the_Lotus_Domino_servlet_engine_on_Windows_qd851

To set up the Domino Servlet Manager perform the following steps within the Domino administrator client:
1. From the Configuration panel, open the server document for the Quickr Domino server.
2. Click Internet Protocols → Domino Web Engine
3. Under Java Servlets, Select Domino Servlet Manager in the Java servlet support field.
4. Save and close the server document.
5. Restart the Domino server and the message "Servlet engine initialization successful" should be displayed.
Set up single sign-on for Quickr Domino

With single sign-on, users can log in to a Quickr Domino server once and, during that session, access all the Quickr places on that server without being prompted to log in again. In the example environment, multi-server single sign-on is configured so that users are able to log in to any server in the environment and then access services from other servers without having to log in again. See Chapter 5, “Implement and test single sign-on” on page 95.

Set up sign-In form for Quickr Domino

Enabling the Quickr sign-in form allows users to be prompted to log in using the Quickr Domino login dialog:

rather than the default Domino server web login dialog:

For detailed information on creating a sign-in form, please refer to the following document http://www-10.lotus.com/ldd/1qwiki.nsf/dx/Completing_single_signon_setup_on_Wind ows_qd851

1. From the Domino Web configuration database (domcfg.nsf), the following values are entered in the sign-in form mapping.
Create a Home URL for Quickr Domino

Creating a home URL on the Quickr Domino server ensures that URLs for the hostname of the server are automatically redirected to the Quickr Domino URL. For example,

http://quickr.demos.ibm.com

is redirected to:

http://quickr.demos.ibm.com/lotusquickr

1. On the server document, **Internet Protocols > HTTP** tab the URL suffix `/lotusquickr` is entered as the Home URL.

Verify the Quickr Domino server configuration

1. Start the Quickr Domino server and enter the following URL into a browser window,

http://quickr.demos.ibm.com/lotusquickr
4.5.3 Configuration

This section details the post-install configuration steps.

Configure LDAP User Directory

In this section the Quickr Domino server is configured to connect to the Domino LDAP server. For more detailed information please refer to the following document:


1. Log in to Quickr Domino using the Quickr administrator ID created during the Quickr Domino installation, In example environment this is "quickradmin"

2. Click on Site Administration → User Directory and select LDAP Server

3. Enter the host name of the LDAP server

4. Click Next and check for the message "OK with Anonymous access"
5. Restart the Domino Server HTTP Task

**Configure qpconfig.xml**

The qpconfig_sample.xml file (located in the Quickr Domino data directory) describes the available settings and their default values, and provides sample xml that you can customize to suit your needs. Please refer to the following link for more information on qpconfig.xml settings:

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Creating_and_using_the_qpconfig.xml_file_advanced_qd851

Make a copy of qpconfig_sample.xml and rename as qpconfig.xml (and always save a copy of your working qpconfig.xml before making new changes!)

To change any of the defaults the comment lines around the section need to be removed in order to make that section live

```
<!-<--------------------------- START OF SAMPLE ------------------------------->
 ......
--------------------------- END OF SAMPLE ------------------------------- -->
```

After changes are made to qpconfig.xml it is good practice to open the file using a browser to check that the format is still correct. If the page displays, and shows the section that was enabled in red, blue and black bold text (instead of grey), then the html formatting is valid.
Customizing qpconfig file for attributes displayed for users and groups

Please refer to the following link on customizing LDAP Directory services
http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Customizing_Lotus_Quickr_control_of_directory_services_qd851

The following steps are carried out to configure the example environment:

1. Open the qpconfig.xml and locate the user_directory section and, within that, the ldap section.

```
<ldap>
  ...
  <base_dn>
    ou=groups,o=ibm
  </base_dn>
  ...
</ldap>
```

14: Serif>
2. Locate the base_dn section and remove the text within the group section. This section should be left blank because Domino groups do not have a hierarchy.

```
<ldap>

 Serif"</base_dn>

 Serif"</group></group>

 Serif"</base_dn>

 Serif">......

 Serif">
```

Restart HTTP task.

**Customizing qpconfig file for search Filters**

Please refer to the follow link on customizing search filters

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Customizing_search_filters_qd851

The following steps are carried out to configure the example environment.

1. Open the qpconfig.xml file and locate the user_directory section, within that, the ldap section and within that the search_filters section.

```
<search_filters>

 Serif"<authentication>

 Serif">![CDATA[

 Serif">[[[(cn=[0])(uid=[0])(shortname=[0])]]]

 Serif"</authentication>

 Serif"></user_lookup>
```

16:  Serif"</![CDATA[

17:
2. Locate the authentication section and append the following text.

   This allows a user to log in to Quickr Domino with their email address as well as common
   name and short name.

   

   <search_filters>
   <authentication>
   
   <![CDATA[
   
   ((cn={0})(uid={0})(shortname={0})
   
   (mail={0}))
   
   <mail={0})]
   
   </authentication>
   
   </search_filters>

3. Locate the user_lookup section and add the following text.

   This allows searches to find matching text in the beginning of both surnames and email
   addresses.

   

   <user_lookup>
   
   <![CDATA[
   
   (&(objectclass=person)(sn={0})(givenname={1}))
   
   (mail={0}))
   
   (mail={0})]]>
   
   </user_lookup>
4. Restart HTTP task.

**Verifying LDAP directory for Quickr Place**
1. To check that user names can be retrieved from the LDAP directory:
2. Log in to Quickr Domino using the credentials of one of the LDAP users.
3. Create a place for the user and go to the Members section.
4. Click **Add Member** and type the first few letters of the first or last name.

![Add members](image)

Authors can read and edit pages. Click OK to add the following users as Author:

- Berzel, Andreas [CN=Andreas Berzel, O=demos]
- Heinz, Betty [CN=Betty Heinz, O=demos]
- Zechman, Betty [CN=Betty Zechman, O=demos]

All matching names should be returned. (Note that the search results contain names where either the first name or the surname begin with the search text).

To check that group names can be retrieved from the LDAP directory:
1. Log in to Quickr Domino using the credentials of one of the LDAP users.
2. Create a place for the user and go to the Members section.
3. Click **Add Member**, click the **Group** radio button and type the first few letters of a group name.
All matching group names should be returned.

### 4.5.4 Additional resources

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Quickr_Domino_8.5_server_installation_and_configuration
Implement and test single sign-on

This section describes implementing and testing single sign-on for our integrated environment.

5.1 What is single sign-on

Single sign-on (SSO) is the ability for a user to log into one server in a single sign-on environment, and then automatically authenticate with other servers in the environment without having to provide login credentials again.

Single sign-on is necessary for an integrated environment, where services from different servers are being exposed within a single application. Without single sign-on, the user would have to complete multiple login screens before all the application functionality became available.

In the example environment, the single sign-on mechanism in use is based on Lightweight Third Party Authentication (LTPA). See the references section at the bottom of the document for links to information on other single sign-on technologies.

For information on using Microsoft Windows single sign-on with Domino and WebSphere Application servers, refer to the following document:

http://www-10.lotus.com/ldd/dominowiki.nsf/dx/Configuring_Microsoft_Windows_single_sign-on_on_IBM_WebSphere_and_Domino_platforms

5.1.1 Single sign-on using LTPA

LTPA, is an single sign-on technology used in IBM WebSphere and Lotus Domino products. A server that is configured to use LTPA authentication will send a session cookie to the browser after successfully authenticating a user. This cookie contains an LTPA token which is valid for
the duration of the browser session, although the token may expire if there is a period of inactivity.

A user with a valid LTPA cookie can access a server that is a member of the same authentication domain as the first server and will be automatically authenticated. The cookie contains the user's distinguished name, the realm to which the user authenticated and a timestamp. All of this information is encrypted and signed using a public/private key pair.

### 5.2 Single sign-on architecture for the example environment

In the diagram below, the user accesses the Portal server and is prompted to log in. The Portal server validates the login credentials provided by the user with the LDAP server and provides the user with an LTPA token. When required, the token is used by the other servers in the environment to validate the user without requiring any input from the user. In this way, the user can access the other servers directly and also services, surfaced inside the Portal server, that are provided by the other servers.

#### 5.3 Prerequisites for single sign-on

The following sections detail the prerequisites for single sign-on to work.

**Common directory**

For simplicity of configuration, all of the servers in a single authentication domain should share the same directory. It is possible to configure single-sign-on if servers in the
environment have different directories but it makes the configuration more complex. Refer to the following document for more information:


The example environment uses a shared directory and this is the Domino LDAP - domino.demos.ibm.com.

**Common DNS domain**
All of the servers in a single authentication domain must be members of the same DNS domain (ie: use the same host name suffix). In the example environment this is .demos.ibm.com.

**Common realm name**
All of the servers in a single authentication domain must use a common realm name. In the example environment this is domino.demos.ibm.com:389

If all of the WebSphere-based servers in the environment use federated repositories then it is possible to leave all the servers configured with the default realm name "defaultWIMFileBasedRealm"

However, in the example environment the Portal server is configured to use the Domino directory as a stand-alone LDAP and as part of this configuration, the realm name is changed to:

**LDAP host name:389**

All other WebSphere Application Servers using a federated repository must have their realm name changed to match this. Note that when configuring the single sign-on setup for Domino servers, the realm name is automatically inherited from the LTPA key when it is imported.

**Time synchronization**
All of the servers in a single authentication domain must be set to the same time. Note that it is not required for all machines to be in the same time zone but they do need to have the same universal time. In the example environment, all the servers are running on virtual machines on a single host and inherit the time from the virtual host.

If single sign-on is only working in one direction (ie: if you log in to server A you can automatically authenticate with server B but, if you log in to server B, you are prompted to log in when you attempt to access server A) it is usually a symptom of a time difference between the two servers.

### 5.4 Implementing single-sign on

This section details the steps taken to configure single-sign-on for the example environment.

For general information on setting up and configuring LTPA please refer to the following document:

http://www-10.lotus.com/ldd/dominowiki.nsf/dx/Configuring_single_sign-on_with_an_LTPA_token_on IBM_WebSphere_and_IBM_Lotus_Domino_platforms
Source server
For simplicity, the LTPA key should be created on a WebSphere server as this key will be accepted by the Domino server with no additional configuration. If the LTPA key is created on a Domino server then additional configuration is required. Refer to the following document for more information:


If there are mixed versions of WebSphere Application Server in the environment it is recommended to create the key using the server on the earliest version. In the example environment, all of the WebSphere servers are using WebSphere Application Server V7 so any of the servers could have created the key.

Create and export the LTPA key
To create and export the LTPA key for the example environment, the following steps were carried out from the Integrated Solutions Console for the Connections server:

The source server in the example environment is the Connections server.

Set realm name
To set the realm name:
1. From Security → Global security → User account repository select Configure

2. In General Properties change the Realm name to domino.demos.ibm.com:389

3. Click Apply and the save link.

Set single sign-on domain
To set the single sign-on domain:
1. From Security → Global Security click Web and SIP security → Single sign-on (SSO) from the Authentication panel
2. Check the box to enable single sign-on (if not already enabled).

3. Enter the domain name for the sample environment .demos.ibm.com. Note that the leading "." is important.

4. Also check that Interoperability Mode is selected.

5. Click **Apply** and the save link.

**Disable auto-generation of keys**

If LTPA is being used to manage single sign-on between the services or components of a single WebSphere cell then allowing the keys to be automatically regenerated at regular intervals provides additional security for the environment. However, when the single sign-on environment includes more than one cell or additional servers as is the case in the example environment, allowing one or more WebSphere servers to auto-generate new keys means the servers will no longer share the same key and single sign-on between those servers will no longer work. The latest versions of WebSphere Application server have this option disabled by default.

To disable the auto-generation of keys:

1. From **Security → Global Security** click **LTPA** in the **Authentication** panel.
2. Click on **CellLTPAKeySetGroup**.

3. Uncheck the option to **Automatically generate keys** (if it is checked).

4. Click **Apply** and the save link.

**Create LTPA key**

To create the LTPA key:

1. From **Security → Global security → LTPA** click **Generate keys**.
2. Click the **save** link.

**Export the LTPA key**

The LTPA key needs to be saved in a file that can be transferred to the other servers that are participating in the single sign-on environment.

To export the key to a file:

1. From **Security → Global Security → LTPA**, in the **Cross-cell single sign-on** panel, enter and verify a password that will be used to encrypt the LTPA key file and provide a name and path to where you want to save the file.

2. Click **Export keys** to save the key file.

**Sync Node**

The node is synchronized to make sure that the changes are inherited by all servers in the node.

1. From **System administration → Save changes to master repository**, select **Synchronize changes with Nodes** and click **Save**.
Clear Scheduled Tasks

Making changes to the LTPA configuration may require running a task to clear any scheduled Connections tasks (they are recreated when the Connections server is restarted). Refer to the following document for more information.

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Clearing_all_scheduled_tasks_ic301

Import key into Domino servers

The following steps were used to import the key into the Domino server configuration and enable single sign-on.

From the server document of the administration server of the Domino domain (Domino/demos):

1. Copy the LTPA key file to the Domino server.
2. Click Create Web... from the action bar and select SSO Configuration.

3. Enter the DNS domain for the example environment .demos.ibm.com and select the Domino servers that will participate in the single sign-on environment. These are the Domino/Notes server (domino/demos) the Sametime Community server (sametime/demos) and the Quickr Domino server (quickr/demos)
4. From the action bar, select **Keys... → Import WebSphere LTPA Keys**.

5. Type in the location and name of the LTPA key file and click **OK**.

6. Enter the password for the key file (as entered when the keys were created on the Connections server) and click **OK** to import the key file.

7. Check the **WebSphere Information** section to see that the correct realm name has been imported.

8. Save and close the document.
Enable single sign-on
The following steps are performed on the server documents of each of the Domino servers participating in the single sign-on environment. In the example environment this is domino/demos, quickr/demos and sametime/demos:

1. From the server document select the Internet Protocols > Domino Web Engine tab.

2. In the HTTP Sessions section, select Multiple Servers (SSO) for the Session authentication and LtpaToken for the Web SSO Configuration.

3. Save and close the server document.

4. Replicate the changes to the other Domino servers.

5. Restart the HTTP task on all the Domino servers.

Import key into Portal server
The following steps were used to import the key into the Portal server configuration and enable single sign-on.

To import the key from the file created on the Connections server:

1. From Security → Global Security → LTPA, in the Cross-cell single sign-on panel, enter and verify the password with which the LTPA keys file was created and provide a name and path to where the file is located.

2. Click Import keys to import the keys into the Portal server configuration.

Enable single sign-on
The following steps are required to complete the single sign-on setup.
1. Check/set realm name.
   The realm name set in the Portal server needs to match the realm name set in the Connections server. In the example environment, the realm name on the Connections server was changed to match the realm name on the Portal server so no change was required. Follow the instructions detailed above to check and/or change the realm name as required. #pointer# to Set realm name section above.

2. Set single sign-on domain. Follow the instructions detailed above to set the single-sign-on domain. #pointer# to Set single sign-on domain section above

3. Check/disable auto-generation of keys. Follow the instructions details above to disable the auto-generation of keys. #pointer# to disable auto-generation of keys section above

4. Restart the Portal server.

**Import key into Sametime servers**
(need to verify with Imran)

The following steps were used to import the key into the Sametime server configuration and enable single sign-on.

To import the key from the file created on the Connections server follow the instructions detailed above #pointer# to Import keys into Portal server.

**Enable single sign-on**
The following steps are required to complete the single sign-on setup.

1. Check/set realm name.
   The realm name set in the Sametime server needs to match the realm name set in the Connections server. Follow the instructions detailed above to check and/or change the realm name as required. #pointer# to Set realm name section above.

2. Set single sign-on domain. Follow the instructions detailed above to set the single-sign-on domain. #pointer# to Set single sign-on domain section above

3. Check/disable auto-generation of keys. Follow the instructions details above to disable the auto-generation of keys. #pointer# to disable auto-generation of keys section above

4. Synchronize the node #pointer# to section above

5. Restart the Sametime servers.

**5.4.1 Testing single sign-on**

From a browser, enter one of the following URLs, login to the server and then switch to any of the other URLs. Regardless of the order of selection, the user should be automatically logged in to all the subsequent servers.
5.4.2 References

Integration overview

This section provides integration overview for the rest of the sections in the wiki.

In the previous sections, we covered the installation of the integrated web environment servers, including:

- 4.1, “Lotus Domino server installation” on page 26
- 4.2, “IBM Sametime installation” on page 31
- 4.3, “IBM Connections installation” on page 53
- 4.4, “Websphere Portal installation” on page 73
- 4.5, “Lotus Quickr for Domino installation” on page 80

To ensure that in the integrated environment, users only have to log in and authenticate once instead of multiple times when accessing various servers, in the previous section, we also covered:

- Chapter 2, “System architecture” on page 17

For integration details, see the following sections:

- Chapter 7, “Portal Integration” on page 109
- Chapter 8, “Connections integration” on page 127
- Chapter 9, “iNotes integration” on page 145
- Chapter 10, “Quickr Domino integration” on page 153
- Chapter 11, “Sametime Integration” on page 161
- Chapter 12, “Creating common navigation across the products” on page 175
Portal Integration

For most users, a Portal server is likely to be the entry point into an integrated web environment. Using portlets, Portal can provide a single page, from which the rest of the features in the integrated web environment can be accessed.

This section describes how Portal is configured to surface functionality from other products in the integrated environment.

7.1 Portal and iNotes integration

This section describes how to configure Portal to surface iNotes functionality.

7.1.1 Overview

Integrating Portal and iNotes is achieved using the Domino Web Access (DWA) portlet. This portlet is installed as one of the out of the box portlets with a Portal installation and it is configured on the Applications -> Messaging -> Mail page.

When using a Lotus Domino LDAP directory, there are two ways to configure the DWA portlet to automatically detect the user's mail file:

1. Configuring automatic mail detection using a Lotus Domino LDAP directory
2. Configuring automatic mail detection using a redirect database

7.1.2 Configuration on Portal

The following configuration changes are necessary on the Portal server:

1. Modify wp_profile_root\portalserver\config\config\CSEnvironment.properties. Ensure the following properties as set according to your environment:
   
   CS_SERVER_DOMINO_DIRECTORY.enabled=true
   
   CS_SERVER_DOMINO_DIRECTORY_1.hostname=domino.demos.ibm.com
110

CS_SERVER_DOMINO_DIRECTORY_1.port=389
CS_SERVER_DOMINO_DIRECTORY_1.ssl=false
CS_SERVER_DOMINO_DIRECTORY_1.anonymous=false
CS_SERVER_DOMINO_DIRECTORY_1.userid=cn=wpsbind,o=demos
CS_SERVER_DOMINO_DIRECTORY_1.encryptedpwd={xor}Lz4sLCgwLTs=

Note: The values shown above are used as an example.

2. Restart Portal.

Notes: If you are following the following wiki article, there are a few mistakes in the documentation:

http://www-10.lotus.com/ldd/portalwiki.nsf/dx/Configuring_a_bind_user_ID_wp7

Step 11. Correction: You must place the file in AppServer_root/bin directory.

Step 18. Correction: There is no dominobind.txt.bak to delete.

They do not specify to set CS_SERVER_DOMINO_DIRECTORY_1.anonymous=false. If this is true, the credentials will not be used and only anonymous LDAP attributes will be able to be read. The portlet needs to find the http-hostname attribute to build the URL to the user's mail file, therefore, anonymous access is not enough.

7.1.3 Configuration on Domino

The following configuration change is necessary on the Domino server:

http://www-10.lotus.com/ldd/portalwiki.nsf/dx/Specifying_the_Internet_host_name_of_the_Lotus_Domino_server_wp7

7.1.4 Verifying the integration

After completing the steps above, the DWA portlet should be working properly as shown below:
7.2 Portal and Connections integration

This section describes how to configure Portal to surface Connections functionality.

7.2.1 Overview

There are two different integration points between Portal and Connections:

1. Connection portlets - There are the following portlets available:
   - Activities -- Collaboration tool for collecting, organizing, sharing, and reusing work related to a project goal.
   - Blogs -- Online journals you can use to deliver timely information with a personal touch.
   - Bookmarks -- Social bookmarking tool that you can use to save, organize, and share Internet and intranet bookmarks.
   - Tag Cloud – Meaningful keywords you can use to find associated content.
   - Profiles – Directory of colleagues you can use to build a network and locate expertise.
   - Wikis – Repository for sharing and collaborating on pages of interest to your group.

2. Business card - This enable the IBM® Connections business card so that your users can access Connections data such as a person's profile information.

This section explains how to configure both integration points.

7.2.2 Prerequisites

One prerequisite for the Connections integration with Portal is the mail attribute must be mapped in Portal. This can be done by following these steps:

1. Edit the following properties in wkplc.properties:
   standalone.ldap.attributes.mapping.ldapName=mail
   standalone.ldap.attributes.mapping.portalName=ibm-primaryEmail

2. Run the following command:
   ConfigEngine.bat wp-update-standalone-ldap-attribute-config -DWasPassword=password

3. Restart Portal

For more information on mapping attributes in Portal, refer to the following documentation:


7.2.3 Required ifixes

In addition to the portal install, you will need to apply the following ifixes to Portal for the integration to work with Connections. All ifixes can be downloaded from fix central.

The first 2 ifixes are installed using the Portal Update Installer.

PM32723
PM52380
The following fix is an update to the connections portlets. This is a war file you use to update the existing connections portlets war file in the Portal Administration user interface.

PM56589

On the connections server, there are updates needed for the business card integration:

LO62605

LO66216

These are also available on fix central and can be applied with the Connections Update installer.

7.2.4 Connection Portlets

Download the Connections portlets
The Connections portlets are not available in the out of the box Portal installation. They are available for download on the portlet catalog. Here is the link:


Install the Connections portlets
All six portlets are packaged in a single war file which can be installed using the Web Modules administration portlet in Portal.

Place the portlets on a Portal page
For the purposes of the demo, a page was created with the following layout:
Assign access to the portlets
Using the Portlets administrative portlet, assign "All Authenticated Portal Users" to the User role for each of the six portlets. If this access is not assigned the portlet will not display on the page for non-admin users.

Import the SSL keys into the Portal server
1. Open the WebSphere Admin Console page of the Portal server.
2. Choose Security →SSL certificate and key management.
3. Choose Key stores and certificates.
4. Select NodeDefaultTrustStore
5. Choose Signer certificates, and click Retrieve from port.
6. Enter values for the Host and Port fields in order to retrieve the SSL certificate as well as an arbitrary alias for the imported certificate. For example:

7. Select Retrieve signer information to retrieve the certificate from the Connections server.
8. Click OK to save the certificate and Save to add this new configuration to the WAS settings.

Configuring portlets to use common directory services
1. On the Connections server, locate the following file:
   \profile_root\config\cells\{local.cell\}/LotusConnections-Config/LotusConnections-config.xml
2. Verify the Profiles database is set up to be the user directory for IBM Connections by looking for the following:

   profiles_directory_service_extension_enabled="true"
3. Locate the elements with the serviceName="communities" and serviceName="profiles" attributes and take note of the values of the and elements as shown below in bold:

```xml
<sloc:serviceReference acf_config_file="acf-config.xml"
bootstrapHost="connections.demos.ibm.com" bootstrapPort="2811" clusterName="LC301"
enabled="true"
person_card_service_name_js_eval="generalsrs.label_personcard_communitieslink"
person_card_service_url_pattern="/service/html/allcommunities?userid={userid}"
serviceName="communities" ssl_enabled="true">        <sloc:href>
<sloc:hrefPathPrefix>/communities</sloc:hrefPathPrefix>            <sloc:static
href="http://connections.demos.ibm.com"
ssl_href="https://connections.demos.ibm.com"/>
<sloc:interService
href="https://connections.demos.ibm.com"/>
</sloc:href></sloc:serviceReference><sloc:serviceReference
bootstrapHost="connections.demos.ibm.com" bootstrapPort="2811" clusterName="LC301"
enabled="true"
person_card_service_name_js_eval="generalsrs.label_personcard_profilelink"
person_card_service_url_pattern="/html/simpleSearch.do?searchFor={userid}&searchBy=userid" serviceName="profiles" ssl_enabled="true">        <sloc:href>
<sloc:hrefPathPrefix>/profiles</sloc:hrefPathPrefix>            <sloc:static
href="http://connections.demos.ibm.com"
ssl_href="https://connections.demos.ibm.com"/>
<sloc:interService
href="https://connections.demos.ibm.com"/>
</sloc:href></sloc:serviceReference>
```

4. On the IBM WebSphere Portal server, copy the following files from
portal_home\wp_profile\installedApps\cell\PA_WPF.ear\snor.pf.portlets.war\WEB-INF\lcel
ator\waltz:
- directory.services.xml
- directory.services.xsd
- sonata.services.xml
- sonata.services.xsd

5. Place a copy of the above files in portal_home\wp_profile\config\cells\cell\

6. Modify the copy of directory.services.xml as shown below:

```xml
<property
name="com.ibm.connections.directory.services.waltz.profiles.integration.service.enabled">true</property><property
name="com.ibm.connections.directory.services.waltz.profiles.integration.service.url">
https://connections.demos.ibm.com/profiles/dsx/</property><property
name="com.ibm.connections.directory.services.waltz.profiles.integration.service.auth">None</property><property
name="com.ibm.connections.directory.services.waltz.profiles.integration.service.auth.alias">connectionsAdmin</property>
```

114
Chapter 7. Portal Integration

For reference, please refer to the directory.services.xml attached to this article as a reference.

7. Restart portal

**Configuring the Portal AJAX proxy to support authentication**

In this step, you will configure the Portal AJAX Proxy to manage authentication for the IBM® Connections portlets. The detailed instructions are documented here:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Configuring_the_Portal_AJAX_proxy_to_support_authentication_ic301

For reference, please refer to the proxy-config.xml attached to this article as a reference.

**Configuring portlets**

After the portlets are installed and placed on a page, you must configure each portlet. To configure the portlet, use the context menu on the top right corner of the portlet and select Configure.

**Configure the Activities portlet**

Here is an example of the configuration for the Activities portlet:

```
Activities
Configuration
Enter the URLs for the IBM Connections servers you want to connect to.

* Service URL:  
https://connections.demos.ibm.com/activities

* Profiles URL:  
https://connections.demos.ibm.com/profiles

* Communities URL:  
https://connections.demos.ibm.com/communities

Display the following Activities Features:  
✔ Show external Connections Links:  
☐ Open links in Connections:  
✔ Display page controls only in bottom bar

Save  Cancel
```

If you get an error while trying to save the configuration for the Activities portlet, please review the following technote:


**Configure the Blogs portlet**

Here is an example of the configuration for the Blogs portlet:
The homepage handle is the homepage set on blogs in step 3 in the following documentation:
http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Configuring_Blogs_ic301

**Configure the Bookmarks portlet**
Here is an example of the configuration for the Bookmarks portlet:

---

**Configure the Tags portlet**
Here is an example of the configuration for the Tags portlet:
Chapter 7. Portal Integration

Configure the Profiles portlet
Here is an example of the configuration for the Profiles portlet:

Configure the Wikis portlet
Here is an example of the configuration for the Wikis portlet:
You can wire any of the IBM Connections portlets in an application. In this demo, we configured the following wires:

7.2.5 Business card

This section describes how to enable the IBM® Connections business card so that your users can access Connections data such as a person's profile information. Please follow these steps:

1. Open the IBM WebSphere Application Server administrative console.
2. In the console navigation tree, select **Resources → URL → URL Providers** and then select the Default URL Provider at the node level of the hierarchy.

3. Under **Additional Properties**, select **URLs**.

4. Click **New** and then specify these settings under **General Properties**:

   ![Integrated Solutions Console](image)

   - **Name**: CONNECTIONS_PEOPLE_CARD
   - **JNDI name**: INDL_CONNECTIONS_PEOPLE_CARD
   - **Specification**: http://connections.demos.ibm.com

   **Note**: The online documentation in the references section has an error in this section. The specification URL should not include /connections.

5. Click **OK** to save the certificate and **Save** to add this new configuration to the WAS settings.

6. Restart server1 and Portal.

### 7.2.6 Verifying the integration

**Verifying the Connections portlets**

After completing the steps above, the Connection portlets should be working properly as shown below:
Verifying the business card

Verify that the IBM Connections business card is enabled, as follows:

1. Open one of the Lotus® Connections portlets on WebSphere Portal and search for a person.

2. Hover on the user's name to make sure the business card displays for this user. If not, the Connections business card is not configured properly.

Here is an example of the business card:
7.2.7 References

For reference, here is a link to the full documentation:

http://www-10.lotus.com/ldd/lcwiki.nsf/dx/Installing_the_Connections_Portlets_for_IBM_WebSphere_Portal_ic301

7.3 Portal and Quickr Domino integration

This section describes how to configure Portal to surface Quickr Domino functionality.

7.3.1 Overview

Integrating Portal and Quickr is achieved using the My Places portlet. This portlet is installed as one of the out of the box portlets with a Portal installation and it is configured on the Applications -> Collaboration-> Lotus Quickr page.

7.3.2 Configuration on Portal

You may download the latest version of the My Places portlet on the Portlet Catalog, however, the latest version does not work properly. Therefore, use the out of the box portlet installed with Portal.
The following configuration changes are necessary on the Portal server:

1. Modify `wp_profile_root/portalserver/config/config/CSEnvironment.properties`. Ensure the following properties as set according to your environment:
   - `CS_SERVER_QUICKPLACE.enabled=true`
   - `CS_SERVER_QUICKPLACE_1.hostname=quickr.demos.ibm.com`
   - `CS_SERVER_QUICKPLACE_1.version=3.0`
   - `CS_SERVER_QUICKPLACE_1.protocol=http`
   - `CS_SERVER_QUICKPLACE_1.port=80`
   
   **Note:** The values shown above are used as an example.

2. Perform the following steps to set the WebSphere Application Server group setting from none to everyone:
   a. Log in to the WebSphere Application Server Administrative Console
   b. Click **Application Types** → **WebSphere enterprise applications** → **PA_MyTeamspaces**.
   c. Click **Security role to user/group mapping**.
   d. Check the **All authenticated users** check box and then select **Everyone** from the **Map Special Subjects** drop-down menu.
   e. Click **OK**.
   f. Click **Save**.
   g. Log out of the WebSphere Application Server Administrative Console.

For reference, see the screenshot below:


### 7.3.3 Configuration on Domino

1. Add the following line to `C:\Lotus\Domino\data\servlets.properties`:
   ```
servlet.QPServlet.initArgs=enablePortal=true
```
Note: Be sure to add a carriage return after the above line.

2. Restart Quickr.

### 7.3.4 Verifying the integration

After completing the steps above, the My Places portlet should be working properly as shown below:

![My Places portlet screenshot](image)

### 7.3.5 References

For reference, here is a link to the full documentation:


### 7.4 Portal and Sametime integration

This section describes how to configure Portal to surface Sametime functionality.
7.4.1 Overview

To configure Portal to surface Sametime functionality, STProxy for awareness and chat is used in the integration. This section provides the steps for the integration.

7.4.2 Prerequisites

1. Install PM42953 on the Portal server. For more information, refer to the following link:
   https://www-304.ibm.com/support/docview.wss?uid=swg1PM42953
2. Install PM52380 on the Portal server.

7.4.3 Installing Lotus Sametime proxy server upgrade for WebSphere Portal V7

The Lotus Sametime proxy server upgrade provides a new Contact List portlet which behaves like the Lotus Sametime web client.

1. Download the portlet from the catalog. Here's the link:
2. Set WasPassword and PortalAdminPwd values in wkplc.properties
3. Run the following two commands on the Portal server:
4. Follow the steps in this technote:
5. Create a page and add the new portlet to a page.

7.4.4 Configuring an HTTP Server as a reverse proxy

The new portlet requires all requests to go through the server you are accessing in the browser session. In this case, that would be the Portal server. There are two options to configure this:

- Using the internal Ajax proxy server in WebSphere Portal
- Using an HTTP Server as your reverse proxy

This section describes using an HTTP server as a reverse proxy.

Configuring a custom property in WAS

1. Open the WebSphere Admin Console page of the Portal server.
2. Click Resources ➔Resource Environment ➔Resource Environment Providers.
3. In the list, locate the WP CommonComponentConfigService and open this entry by clicking it.
4. Click Custom properties.
5. Locate the entry cc.sametime.proxy.url. If an entry with this name does not exist yet, create it.

6. Specify a "/stproxy" as the value for cc.sametime.proxy.url as shown below:

7. Restart Portal.

**Configuration on the HTTP Server**

To configure HTTP Server:

1. Edit httpd.conf

   Uncomment the following:

   LoadModule proxy_module modules/mod_proxy.so
   LoadModule proxy_http_module modules/mod_proxy_http.so

   Add the following:

   ```
   1:
   2:  <IfModule mod_proxy.c>
   6:      ProxyPassReverseCookiePath / /stproxy
   7:  </IfModule>
   8:
   ```

2. Restart IHS.
7.4.5 Configuration on Sametime

The following documentation provided the detailed configuration steps required on the Sametime server:

1. Changing the session cookie ID on the Lotus Sametime proxy server: V7, please refer to this link:
   

2. Integrating the functionality of the Sametime Proxy into Websphere Portal, please refer to this link:
   

7.4.6 Verifying the integration

After completing the steps above, the Sametime portlet should be working properly as shown below:

![Sametime Web 2.0 Contact List](image)

Please note that you will see the following error in the admin pages:

![Message from webpage](image)

This is a known issue and a fix is not available at the time of writing.
Connections integration

This section describes configuration steps required to support the surfacing of Connections functionality within the other products in the example environment and provides links to the documents where this integration is configured. Links are also included to the documents that describe how Connections can surface functionality from other products in the environment.

This Connections integration is covered from the following perspectives:

- Connections as a provider - This describes the integration elements where Connections features and functionality are surfaced within another product:
  - Connections integration within Sametime: See 12.1, “Customizing the WebSphere Portal theme” on page 176 on how to configure Sametime to surface Connections functionality.

- Connections as a consumer - This describes the integration elements where Connections surfaces the functionality of another product:
  - Sametime integration within Connections: See 8.1, “Connections and Sametime integration” on how to configure Connections to surface Sametime functionality.
  - Quickr Domino integration within Connections: See 8.2, “Connections and Quickr Domino integration” on page 131 on how to configure Connections to surface Quickr Domino functionality.

8.1 Connections and Sametime integration

This section describes integration of Connections and Sametime.

When adding Sametime capabilities to Connections, there are two primary ways to do this, both involve adding chat and awareness capabilities to usernames throughout Connections:
First, it requires you to have the full Sametime 8.5.x client installed on the users local machine.

Second, it requires no client installation on the machine; instead, it uses the capabilities of the STProxy 8.5.1.1 server for chat and awareness. In addition this approach checks if a Sametime client is installed on the local machine and use that client to start a chat to give the end user the most capabilities possible when chatting with another user.

The following subsection shows you how to setup and configure the second option:

### 8.1.1 Connections integration with STProxy for chat and awareness

This section describes Connections integration with STProxy for chat and awareness.

**Overview**

Using the Sametime Proxy 8.5.1.1 server you can enable presence awareness and simple chats in IBM Connections without having to download and install a full sametime client. To configure this, perform the following steps.

**Configure SSO between Connections and the Sametime Community Server**

Configure SSO between Connections and the Sametime Community Server as described in Chapter 5, “Implement and test single sign-on” on page 95.

Note: There is no need to add the key file to the STProxy WebSphere server. There is no check-in of the user name or LtpaToken by the STProxy server. It simply forwards the request to the community server to authenticate and login the user.

**Configure the IBM HTTP Server in front of Connections to proxy requests to STProxy**

The ajax requested used by STProxy in 8.5.1.1 require all request go through the server you are accessing in the browser session. In this case that would be the Connections server hostname. Therefore, we must configure the HTTP Server in front of Connections to act as a proxy for awareness and chat communication between the client and STProxy server.

The following steps configure this on the HTTP Server.

1. Open E:\IBM\HTTPServer\conf\httpd.conf in a text editor (we used notepad)
2. Search for and uncomment the following lines:
   
   ```
   LoadModule proxy_module modules/mod_proxy.so
   LoadModule proxy_connect_module modules/mod_proxy_connect.so
   LoadModule proxy_http_module modules/mod_proxy_http.so
   ```

3. Add the following lines to the file (Note: I added these just above the following lines at the bottom of the file:)
   
   ```
   LoadModule was_ap22_module
   "E:\IBM\HTTPServer\Plugins\bin\32bits\mod_was_ap22_http.dll"
   WebSpherePluginConfig "E:\IBM\HTTPServer\Plugins\config\webserver1\plugin-cfg.xml"
   ```

   DO NOT add them below the WebSpherePluginConfig line
proxy rules to add:

ProxyPassReverseCookiePath //

Note: Replace stproxy.demos.ibm.com:9081 with the hostname:port of your stproxy server.

4. Save and close httpd.conf.

5. Restart the IBM HTTP Server.

Test IHS connectivity to STProxy Server
To test IHS connectivity to STProxy Server, follow these steps:

1. Open a browser to STProxy using the Connections HTTP server: port (https://connections.demos.ibm.com/stwebclient/index.jsp)

2. You should see the default Sametime web client page.

Enable Connections configuration with STProxy
To enable Connections configuration with STProxy:

1. Open a command prompt to the dmgr bin directory (E:\IBM\WebSphere\AppServer\profiles\Dmgr01\bin)
2. Run the following command
   
   wsadmin -lang jython -user wasadmin -password password -port 8879

3. At the wsadmin prompt run the following commands:
   
   execfile("connectionsConfig.py")
   
   LCConfigService.checkOutUilextensionsConfig("e:/temp", "connectionsCell01")
   
   Where e:/temp is a temp directory on the local machine and connectionsCell01 is the cell for the Connections environment.
   
   Note: to verify the cell name, run the following command at the wsadmin prompt: print AdminControl.getCell()

4. Leave the wsadmin prompt and window open

5. Open uilextensions-config.xml file from the temp directory in a text editor, we used notepad, scroll down near the bottom of the file and look for the following lines:

```
1: 2: 3: <extension name="global.pageRendering" jsCallback="lconn.profiles.sametime.sametimeProxyInit" enabled="false"/>
4:   <extension name="lc.IMAwareness" jsCallback="lconn.profiles.sametime.sametimeProxyAddLiveName" invokeDelay="2000" enabled="false" id="stProxy">
5:     <params>
6:       <param name="isConnectClient" value="true" />
7:       <param name="lconnProxySvcUrl" value="admin_replace"/>
8:       <param name="lconnProxySvcUrlSSL" value="admin_replace"/>
9:     </params>
10:   </extension>
```

Update this section by setting enabled="true" and setting the lconnProxySvcUrl to the hostname of the HTTP Server used to access Connections

```
1: 2: 3: <extension name="global.pageRendering" jsCallback="lconn.profiles.sametime.sametimeProxyInit" enabled="true"/>
4:   <extension name="lc.IMAwareness" jsCallback="lconn.profiles.sametime.sametimeProxyAddLiveName" invokeDelay="2000" enabled="true" id="stProxy">
5:     <params>
6:       <param name="isConnectClient" value="true" />
7:       <param name="lconnProxySvcUrl" value="http://connections.demos.ibm.com"/>
8:       <param name="lconnProxySvcUrlSSL" value="https://connections.demos.ibm.com"/>
9:     </params>
10:   </extension>
```
6. Save and close uiextensions-config.xml.
7. Back in the wsadmin command window, enter the following command:
   LCConfigService.checkInUIextensionsConfig("e:/temp", "connectionsCell01")
8. Type exit to exit out of the wsadmin console.
9. Restart the Connections server:
   a. Open a browser to the WebSphere® Application Server Integrated Solutions Console
      for the Deployment Manager and sign in
   b. Click Servers -> Server Types -> WebSphere application servers.
   c. Select the Connections server (LC301_server1) and click Restart.

Test awareness and chat in Connections
To test the awareness and chat in Connections:
2. Sign in as a test user (wpsadmin : password).
3. At the bottom right corner of the browser you should see the user show as available:
4. Also, if you search for the user, and view their business card, you will see the user as
   Online, and have an option to start a chat with the user.

8.2 Connections and Quickr Domino integration
This section describes Connections and Quickr Domino integration.
When adding Quickr capabilities to Connections there are two primary ways to do this, as explained in the following sections.

8.2.1 Communities integration with Quickr Domino

Overview
With the IBM® Connections Connector for Lotus® Quickr™, you can create associated Lotus Quickr team places for IBM Connections communities. A Lotus Quickr team place provides a central location from which a team can organize, share files, and collaborate on documents. Every IBM Connections community owner can optionally choose to create an associated Lotus Quickr team place. Updates made in the Lotus Quickr team place are aggregated in the community overview page, making it easier to stay current with projects and work collaboratively. Access control and membership of the Lotus Quickr place is determined by the often fluctuating IBM Connections community membership; it does not need to be managed separately.

Configure single sign-on between Connections and Quickr Domino
To configure single sign-on between Connections and Quickr Domino, refer to Chapter 5, “Implement and test single sign-on” on page 95.

Configure admin user in Quickr Domino
During the install and setup of the Connections Connector for Lotus Quickr you have to specify a user that the Connections server will use to authenticate with the Quickr server to create the Quickr place and keep the members of the community and Quickr place synchronized. This user must be an LDAP user that has access to Connections, as well as an Administrator of the Quickr Domino server. The reason the user must exist in LDAP, and the Connections server is because when we make calls to Quickr, the connector will get the username / password of the connector user, call into WebSphere with that name and password. WebSphere will authenticate the user and return an LtpaToken to the connector, then the connector will make the request to Quickr sending the LtpaToken. So the default Quickr admin configured during the Quickr install will not work in this setup as that user is only known by the Quickr server, and not Connections.

Use the following steps to add a user to the Administrator role on the Quickr Domino Server:
1. Open a browser to the Quickr server (http://quickr.demos.ibm.com/lotusquickr) and sign in as the Quickr Administrator (qpadmin : password).
2. Click Site Administration.
3. Click Security.
4. Under Who can Administer this server, click Add.
5. Click Directory.
6. Search for the user in LDAP you would like to add.
7. Select the user and click Add, you should see a Successfully Added: username message, Click OK.
8. Click Close in the Directory Search window.
10. You should see the user in the list (wpsadmin in our example):
Download and install IBM Connections Quickr Connector 3.0.1

To download and install IBM Connections Quickr Connector 3.0.1:

1. Download the IBM Connections Quickr Connector 3.0.1 from the portlet catalog to the Connections Server.
   
   https://greenhouse.lotus.com/plugins/plugincatalog.nsf/assetDetails.xsp?action=editDocument&documentId=C13366D0C29ADFCC8525786A003BFD06

2. Unzip this to a temp directory on the Connections server.

3. Run Lotus_Connections_Connectors_Quickr_3.0.1_win to unzip the install files to the temp directory.

4. Run LC_Connectors_Quickr_Install_IM\IM\windows\install.bat.

5. On the IBM Install Manager page, select IBM Lotus Connectors for Quickr and Version 3.0.1.0 and click Next.
   
   NOTE: If you see the IBM Install Manager page requested you to click Install, or Properties, you ran install.exe, close the IBM Installation Manager and run install.bat.
6. Review and accept the license agreement by selecting I accept the terms in the license agreements. Then click **Next** to continue.

7. Specify the location where you want to install the connector in the **Installation Directory** field (E:\IBM\ConnectorsQuickr), and then click **Next**.

8. Make sure IBM Lotus Connectors for Quickr and Version 3.0.1.0 are selected, and then click **Next**.

9. Specify Quickr type: Quickr Domino:
   - The templates you want to use from the Quickr Domino server
   - Hostname and port of the Quickr Domino server
   - Finally, the J2C authentication user name and password should be the user in LDAP you added as a Quickr administrator earlier.
10. Enter the Lotus Connections install home directory, this should cause the Connector libraries install location and configuration install location to pre-populate, and click Validate.
11. Click **Next**.

12. Click **Install**.

13. Once the install completes you will see:

   Install results :
   
   Installed successfully.

14. Click **Finish**.

**Synchronize the deployment manager with the nodes**

To complete the install of the IBM Connections Quickr Connector you need to synchronize the DMGR with each node.

1. Open a browser to the WebSphere® Application Server Integrated Solutions Console for the Deployment Manager and sign in
   

2. Select **System administration -> Nodes**, select the Connections nodes, and then click
   
   **Full Resynchronize**.
3. From the main Integrated Solutions Console page, select Servers ➔ Clusters ➔ WebSphere Application Server Clusters.

4. Select the check box beside the cluster containing the nodes that have been updated, and then click Stop.

5. Once the cluster is completely Stopped Select the cluster and click Start.

Add the Quickr Domino server as a supported server in Connections

We provided the IBM Lotus Quickr server to the ajaxproxy server to ensure that it honors any requests made for access to one of the supported Lotus Quickr servers:


2. Expand Resources ➔ Resource Environment, and then click Resource Environment Providers.

3. Click QuickrWhitelistProvider from the list, and then click Custom properties.

4. Click New.

5. Enter a name that starts with allow (eg: "allow" and the host name of the server -- allowquickr.demos.ibm.com) and the value should be the hostname of the Quickr server.
6. Click **OK**.

7. Click **Save** to save the changes to the master configuration.

8. Once the change is automatically synchronized with the nodes, click **OK**.


**Configure Connections ajaxproxy to work with Quickr**

Communication between Connections and Quickr goes through the IBM Connections ajaxproxy. By default it is configured to allow cookies, headers or mime types, and all HTTP actions to be exchanged among the IBM Connections applications. However, it also prevents HTTP GET requests from non-IBM Connections services and prevents all cookies or headers from being directed to the applications.

We need to configure the Connections ajaxproxy to allow HTTP GET requests to the Quickr server, and send the LtpaToken for authentication:

1. Open a command prompt to the dmgr bin directory
   (E:\IBM\WebSphere\AppServer\profiles\Dmgr01\bin).
2. Run the following command:
   `wsadmin -lang jython -user wasadmin -password password -port 8879`

3. At the wsadmin prompt run the following commands:
   ```python
   execfile("connectionsConfig.py")
   execfile("communitiesAdmin.py")
   LCConfigService.checkOutProxyConfig("e:/temp", "connectionsCell01")
   ```
   Where e:/temp is a temp directory on the local machine and connectionsCell01 is the cell for the Connections environment.
   Note: to verify the cell name, run the following command at the wsadmin prompt: `print AdminControl.getCell()`

4. Leave the wsadmin prompt and window open.

5. Open proxy-config.tpl file from the temp directory in a text editor, we used notepad, scroll down near the bottom

6. Add the following rules for your Quickr server:

   ```xml
   <proxy:policy url="http://quickr.demos.ibm.com/*" acf="none">
   <proxy:actions>
   <proxy:method>GET</proxy:method>
   </proxy:actions>
   <proxy:headers>
   <proxy:header>User-Agent</proxy:header>
   <proxy:header>Accept*</proxy:header>
   <proxy:header>Content*</proxy:header>
   <proxy:header>Authorization*</proxy:header>
   </proxy:headers>
   <proxy:cookies>
   <proxy:cookie>JSESSIONID</proxy:cookie>
   <proxy:cookie>LtpaToken</proxy:cookie>
   <proxy:cookie>LtpaToken2</proxy:cookie>
   </proxy:cookies>
   </proxy:policy>
   ```

7. Save and close the proxy-config.tpl.

8. Back in the wsadmin command window, enter the following command:
   `LCConfigService.checkInProxyConfig("e:/temp", "connectionsCell01")`

9. Type `exit` to exit out of the wsadmin console.

10. Restart the Connections server:
    a. Open a browser to the WebSphere® Application Server Integrated Solutions Console for the Deployment Manager and sign in
    b. Click **Servers** → **Server Types** → **WebSphere application servers**.
    c. Select the connections server (LC301_server1) and click **Restart**.

**Test Communities Quickr integration**

To test Communities Quickr integration:

2. Sign in as a test user (wpsadmin : password).

3. Click **Start a Community**.

4. Enter a name, and under Associated Applications, select Include this application in the community: Quickr Teamspace.

5. Click **Save**.

6. The Community and Quickr place should be created, after they are created, the community will appear with a Quickr Teamspace widget showing the home page of the place.

---

**8.2.2 Activities integration with Quickr Domino**

This section describes Activities integration with Quickr Domino.

**Overview**

After you have finished collaborating with your team on a file in an activity, you can publish the finished product to an IBM Lotus Quickr library to share it with a larger group or to store the file.

For this integration to work you need to:

1. Enable SSO between Connections and Quickr

2. Edit the configuration file to enable publishing a file attachment to a Quickr library

3. Define a list of Quickr servers so all necessary cookies are sent from Activities to the Quickr server.

**Enable SSO between Connections and Quickr**

To enable single sign-on between Connections and Quickr, follow the steps in Chapter 5, “Implement and test single sign-on” on page 95.

**Enabling users to publish file attachments to Lotus Quickr**

To enable users to publish file attachments to Lotus Quickr:

1. Open a command prompt to the dmgr bin directory (E:\IBM\WebSphere\AppServer\profiles\Dmgr01\bin).

2. Run the following command:
   ```bash
   wsadmin -lang jython -user wasadmin -password password -port 8879
   ```

3. At the wsadmin prompt, run the following commands:
   ```python
   execfile("activitiesAdmin.py")
   ```
ActivitiesConfigService.checkOutConfig("e:/temp", "connectionsCell01")
Where e:/temp is a temp directory on the local machine and connectionsCell01 is the cell for the Connections environment.

Note: to verify the cell name, run the following command at the wsadmin prompt: print AdminControl.getCell()

4. Leave the wsadmin prompt and window open.

5. Open oa-config.xml file from the temp directory in a text editor, we used notepad; search in the file and look for the following section.

```xml
<PublishFile enabled="false" allowCustomServers="false" requireSSO="true">
    <server>http://localhost:8080</server>
    <server>http://localhost:8080</server>
</PublishFile>
```

Updated it to match your environment. We used the following:

```xml
<PublishFile enabled="true" allowCustomServers="false" requireSSO="true">
    <server>http://quickr.demos.ibm.com</server>
</PublishFile>
```

6. Save and close the file.

7. Back in the wsadmin command window, enter the following command:
   ActivitiesConfigService.checkOutConfig("e:/temp", "connectionsCell01")

8. Type exit to exit out of the wsadmin console.

You will need to restart the Connections server for these changes to take effect, but we will first define this server in WebSphere so all necessary cookies are sent from Activities to the Quickr server.

**Add Quickr server to list of supported servers in WebSphere**

Note: If you configured Communities with this Quickr server, you have already completed this step. In which case, skip to step 10 to restart Connections and test that it works in your environment.


3. Click QuickrWhitelistProvider from the list, and then click Custom properties.

4. Click New.

5. Enter a name that starts with allow (e.g. "allow" and the hostname of the server -- allowquickr.demos.ibm.com) and the value should be the host name of the Quickr server.
6. Click OK.
7. Click Save to save the changes to the master configuration.
8. Once the change is automatically synchronized with the nodes, click OK.


10. Restart the Connections server:
    b. Click Servers → Server Types → WebSphere application servers.
    c. Select the Connections server (LC301_server1) and click Restart.

**Test publishing a file from Activities to a Quickr library**

To test, publishing a file from Activities to a Quickr library:
1. Open your browser to activities. If you do not have an activity create a new one and upload a file to an entry.
2. Click More next to the entry.
3. Next to where you can download the attachment, click Publish to Lotus Quickr.
iNotes integration

This section describes configuration steps required to support the surfacing of iNotes functionality within the other products in the example environment and provides links to the documents where this integration is configured. Links are also included to the sections that describe how iNotes can surface functionality from other products in the environment.

This iNotes integration is covered from the following perspectives:

- **iNotes as a Provider -** This describes the integration elements where iNotes features and functionality are surfaced within another product:
  - iNotes integration within Portal: See 7.1, “Portal and iNotes integration” on page 109 on how to configure Portal to surface iNotes functionality.

- **iNotes as a consumer -** This describes the integration elements where iNotes surfaces the functionality of another product:
  - Sametime integration within iNotes: See 9.1, “iNotes -- Sametime integration” on how to configure iNotes to surface Sametime functionality.
  - Quickr Domino integration within iNotes: See 9.2, “iNotes -- Quickr Domino integration” on page 148 on how to configure iNotes to surface Quickr Domino functionality.

### 9.1 iNotes -- Sametime integration

This section describes how to configure iNotes to surface Sametime functionality.

#### 9.1.1 Enable Sametime awareness in iNotes

Domino 8.5.3 allows integration with the Sametime server using the Sametime Proxy 8.5.2 external link for Web 2.0 user experience which greatly simplifies the Domino HTTP and iNotes configuration and enhances the interface and features available to users.

**Enable Sametime awareness in the server configuration document**

In order to configure Sametime awareness to be an available option for iNotes, complete the following steps:
1. Create a configuration document for the server on which iNotes runs.

2. Click the Lotus iNotes tab of the Configuration document and complete the following fields.

3. Save and close the configuration document.

4. In the Domino console enter `restart task http` to enable this feature on the server.

Enable Sametime awareness in the user mailfile

Each user needs to enable Sametime in their preferences as follows:

1. Login to iNotes. For direct access to a mail file via iNotes, type the following into the browser `http://hostname/mail/mailfile name`.

2. Click on the Preferences tab, scroll down and click Enable Instant messaging.
If you do not use Domino 8.5 and the Sametime proxy refer to the Enable the Instant Messaging documentation online.

### 9.1.2 Testing Sametime awareness

To validate the configuration, complete the following steps:

1. Login to iNotes as a user and open the Inbox.
2. When the screen has refreshed, the Sametime status icon beside the user name in the left navigator should appear green.
3. Check the Inbox and note that the Sametime status of users listed in the **Who** field is displayed.
4. Click on the arrow beside the status icon and select **Show Sametime Contact List** and the Sametime status of contacts is displayed.
5. Right click on a person's name in the Inbox or Sametime Contacts list to start a chat or meeting.
9.2 iNotes -- Quickr Domino integration

This section describes how to configure iNotes to surface Quickr Domino functionality.

9.2.1 Overview

iNotes integration with Quickr Domino enables users to:

- Add a Quickr Domino place from iNotes user preferences.
- Choose whether to store an attachment in a Quickr place and send a link to the attachment rather than send the attachment itself within an email.
- Save an attachment from a received mail into a Lotus Quickr place.
- Access files stored in Quickr Domino places from links in messages.

For more detailed information please refer to the following document on "Integrating Lotus iNotes with Lotus Quickr for Domino"

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Integrating_Lotus_iNotes_with_Lotus_Quickr_for_Domino_8.5.1

9.2.2 Create Policy settings to enable integration with Quickr

To allow iNotes users to access Quickr places and resources from the iNotes mail file, a policy external link is defined on the iNotes server.

The following steps describe the policy configuration to authorises access to the Quickr server in the example environment from the iNotes server in the example environment.

1. From the Domino administrator client create an appropriately named organizational policy e.g. */demos. This will allow all users in the organization to use this feature.
2. Create a mail setting for the policy to allow users to choose Quickr places and send attachments to a Quickr place while sending addressees a link.
   a. Give the setting an appropriate name e.g. iNotes-Quickr Mail.
   b. Click the iNotes tab.
   c. Select Enable in the Allow Quickr integration field.
   d. Save the setting.

3. Create a security setting for the policy to allow Domino to securely send user requests and communicate with Quickr server. For more detail refer to the following document: http://www-10.lotus.com/ldd/lqwiki.nsf/dx/iNotes_proxy_configuration_qd851
   a. Give the setting an appropriate name e.g. iNotes-Quickr Security.
   b. Click the Proxies tab.
   c. Enter the following to create the whitelist rule.
4. Restart the Domino server to enable both of these policies.

9.2.3 Enable Quickr preferences in iNotes

Each user will need to enable these preferences individually:

1. From within iNotes, click on Preferences and select Lotus Quickr.

2. Click Add Places...

3. Enter the host name of the Quickr server in the dialog box (not the URL) and click Go. The list of places of which the user is a member should appear.
4. Select the Places to be added and click **OK**.

5. Select option for sending attachments and a default Place to save attachments. In this example **Prompt me** is selected.

6. Save and close preferences.

### 9.2.4 Test Quickr integration

To validate the configuration:

1. Send an email with an attachment to a member of one of the user's Quickr places (Note: Recipients must be members of the Quickr place where the attachment is stores or they cannot access the file).

2. When prompted choose to save the file to a place and send the email recipient a link.
3. The recipient receives the email with the link to where the attachment has been posted.
4. The recipient clicks on one of the links associated with the attachment to either download or open the summary page for the attachment.
Quickr Domino integration

This section describes configuration steps required to support the surfacing of Quickr Domino functionality within the other products in the example environment and provides links to the documents where this integration is configured. Links are also included to the documents that describe how Quickr Domino can surface functionality from other products in the environment.

This Quickr Domino integration is covered from the following perspectives:

► Quickr Domino as a Provider - This describes the integration elements where Quickr Domino features and functionality are surfaced within another product:
  – Quickr Domino integration within Portal: See 7.3, “Portal and Quickr Domino integration” on page 121 on how to configure Portal to surface Quickr Domino functionality.

► Quickr Domino as a consumer - This describes the integration elements where Quickr Domino surfaces the functionality of another product:
  – Sametime integration within Quickr Domino: See 10.1, “Lotus Quickr for Domino and Sametime integration” on how to configure Quickr Domino to surface Sametime functionality.

10.1 Lotus Quickr for Domino and Sametime integration

This section describes how to configure Quickr Domino to surface Sametime functionality.

10.1.1 Overview

Quickr Domino can be integrated with Sametime to take advantage of Sametime awareness and instant messaging in Quickr places. The members of a Quickr Domino place can see when other members are online and chat with other members from within the place.
Please note that the latest version of Quickr Domino available at the time of writing is Quickr Domino 8.5.1 which does not have support for the Sametime 8.5 web client. Therefore the Quickr Domino integration with Sametime detailed here is based on the use of the STLinks capabilities from the Sametime SDK.

### 10.1.2 Preparing the Quickr and Sametime servers for awareness and Instant messaging

Certain files have to be copied to the Sametime server to enable Quickr Domino to make use of Sametime awareness and instant messaging. This section describes where to locate these files and where to place them.

2. Download the Sametime 8.5.2 SDK onto the Sametime Server and extract the files.
3. Create a sub folder on Sametime server as follows:
   
   **Data Directory\Domino\html\QuickPlace\peopleonline**

4. Copy the following files from the `client\stjava\bin` subdirectory of the Sametime SDK to the above directory on the Sametime server.
   
   - STComm.jar
   - CommRes.jar

5. Copy the following file from the **Data Directory\LotusQuickr** directory of the Quickr Domino server to the above directory on the Sametime server.
   
   - PeopleOnline31.jar

### 10.1.3 Configuring the Sametime server for Quickr Domino server

The Quickr Domino server is configured with the location of the Sametime server using the Site Administration feature on the Quickr Domino server.

1. Log in as the administrator
2. Click on **Site Administration** in the footer of the Welcome Page.
3. Click on Other Options and then Edit Options.

4. In the Sametime Servers section enter the URL to the Sametime community server. In the example environment, this is sametime.demos.ibm.com.

5. Click Next to finish.


10.1.4 Verifying configuration

Complete the following steps to verify that Sametime awareness and instant messaging is working correctly:

1. Log in to Quickr server using any LDAP user credentials.

2. Check for the green icon next to the login name which indicates that Sametime awareness is enabled.
3. Click on one of the places and notice that author names also show presence awareness.
4. Make sure that one of the document authors is online and click on the green icon next to their name to open an instant message chat window.

10.1.5 References

Please refer to the following document for more details on integrating Quickr Domino and Sametime for awareness and Instant messaging.

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Enabling_Lotus_Sametime_features_in_places_qd851

10.2 Lotus Quickr for Domino and Lotus Connections integration

This section describes how to configure Quickr Domino to surface Connections functionality.

10.2.1 Overview

Lotus Quickr Domino supports the integration of information stored in Connections in two ways:
Connections Profile business card can be displayed wherever there are live names in Quickr Domino.

Photos stored in Connections can be displayed in:

- Quickr Domino place member profiles
- Forum postings
- Comment documents

Integrating Quickr Domino with Connections is achieved using qpconfig.xml file. The qpconfig.xml file is used to perform many server configuration and integration tasks carried out by an administrator. The Quickr Domino server comes with a sample file, "qpconfig_sample.xml", which is stored in Quickr Domino server's data directory. For more detailed information please refer to the following document on creating and using qpconfig.xml file.

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Creating_and_using_the_qpconfig.xml_file_advanced_qd851

10.2.2 Configuring Quickr Domino to display photos from Connection profiles

This section describes how to configure Quickr Domino to display photos, that have been stored in Connections profiles, in any photo field in Quickr. This includes Quickr Domino member profiles, forum postings and comments fields. Note that this integration feature is not restricted to photos stored in Connections and can be used with any photo store that offers a similar web service for accessing the photos.

Verifying web service request

The format of the Connections web service request is http://Connections server hostname/profiles/photo.do?email={email} where {email} is the email address of the user whose photo is to be returned.

To verify that this will return a photo from the Connections server:

1. Enter the URL into a browser window, replacing {email} with a valid user email address.
   - e.g. http://connections.demos.ibm.com/profiles/photo.do?email=ed.blanks@demos.ibm.com

2. Verify that the browser displays the expected photo.
Configuring Quickr Domino to use web service request

In order to display photos from Connections in Quickr Domino photo fields, the web service request is added to the user_directory section in qpconfig.xml. To do this:

1. Open the qpconfig.xml file in the Quickr Domino server's data directory. Note: if there is no qpconfig.xml in the directory make a copy of the qpconfig_sample.xml file and name this qpconfig.xml.

2. Go to User Directory section and enter the web service call that returns user photos from Connections profiles as show below.

```xml
1:   <user_directory>
2:     <user_photo_source>
3:       <directory>
4:         <url>
5:           <![CDATA[
7:           ]]>}
8:       </url>
9:     </directory>
10:   </user_photo_source>
```

3. Restart HTTP task.

Verify the display of photos in Quickr Domino

In order to validate the display of photos from Connections profiles in Quickr Domino log in to Quickr Domino and open a place. The configuration is successful if photos are displayed in:

- Member profiles
10.2.3 Configuring Quickr Domino to display Connections Business Cards

In order to display business cards from Connections wherever there are live names in Quickr Domino, information is added to the profile_server section in qpconfig.xml. To do this:

1. Open the qpconfig.xml file in the Quickr Domino server's data directory. Note: if there is no qpconfig.xml in the directory make a copy of the qpconfig_sample.xml file and name this qpconfig.xml.

2. Go to profile_server section enter:
   – the host name of the Connections server
   – the location of the semantic tag service

```
<profile_server>
  <server_name ssl="false">
    connections.demos.ibm.com
  </server_name>
  <semantic_tag_service_location>
    /profiles/ibm_semanticTagServlet/javascript/semanticTagService.js
  </semantic_tag_service_location>
</profile_server>
```
3. Restart HTTP task.

**Verify the display of Connections business cards in Quickr Domino**

In order to validate the display of Connections business cards in Quickr Domino, log in to Quickr Domino and open a place. The configuration is successful if:

- Connections business card links are available wherever there are live names

- Clicking on a business card link retrieves the business card photo for the associated live name

10.2.4 References

For reference, here is a link to the full documentation:

http://www-10.lotus.com/ldd/lqwiki.nsf/dx/Enabling_Lotus_Connections_features_in_Lotus_Quickr_qd851
Sametime Integration

This section details the configuration steps required to support the surfacing of Sametime functionality (presence awareness and instant messaging) within the other products in the example environment and provides links to the documents where this integration is configured. Links are also included to the documents that describe how Sametime can surface functionality from other products in the environment.

11.1 Overview of Sametime integration

At the time of writing the only product in the example environment that can make use of the Sametime 8.5.2 proxy server is iNotes. Quickr Domino 8.5.1 does not yet have support for any version of the Sametime Proxy server and uses the traditional stlinks capabilities to connect to the Sametime Community server to provide presence awareness and instant messaging within Quickr places. Portal and Connections support the Sametime 8.5.1.1 Proxy server (with some additional fixes). Therefore to provide presence awareness and instant messaging for all the products in the example environment, an additional Sametime Proxy server (8.5.1.1) is installed alongside the Sametime Meeting server.

The diagram shows the pieces of installation and configuration that will be discussed in this chapter highlighted in red.
The dotted lines in the diagram indicate the communication paths used for configuring the system.

The unbroken lines in the diagram indicate the communication paths used in operating the system.

11.2 Sametime as a provider

This section details the integration elements where Sametime features and functionality are surfaced within another product.

At the time of writing, Portal (7.0.0.1) and Connections (3.0.1) do not support the Sametime 8.5.2 proxy and iNotes (8.5.3) does not support anything except the Sametime 8.5.2 proxy. Therefore, for the example environment, both proxy servers are deployed. The Sametime 8.5.2 proxy server is installed as part of the System Console cell (#see Chapter 4.4#) and the Sametime 8.5.1.1 proxy server is installed as a stand-alone server which can easily be removed from the environment when later versions of Portal and Connections support the Sametime 8.5.2 proxy.

11.2.1 Prepare Sametime to support instant messaging and presence awareness in Portal and Connections

Download the following part numbers to install the required Sametime components:
Install Sametime 8.5.1 Proxy server

For the example environment, the Sametime 8.5.1 proxy server is installed on the same server as the Sametime Meeting server. The following options are selected during the Sametime 8.5.1 proxy server installation:

1. Deselect the option to Use Lotus Sametime System Console to install.

2. Complete the rest of the installation screens using the entries in the table below.

<table>
<thead>
<tr>
<th>Part Numbers</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CZYE0ML</td>
<td>IBM Lotus Sametime Standard V8.5.1 Proxy Server Windows Multilingual</td>
</tr>
<tr>
<td>BMUY-8F9PUJ (ifix)</td>
<td>8.5.1.1-Lotus-ST-FP1-SametimeStandard Server - ST Proxy (download from Fix Central)</td>
</tr>
<tr>
<td>WHOS-8MAJ6R (hotfix)</td>
<td>sametimeproxy.ear (as referenced in <a href="http://www-01.ibm.com/support/docview.wss?uid=swg21509230">http://www-01.ibm.com/support/docview.wss?uid=swg21509230</a> - needs to be requested from IBM Sametime Support. This file is used in place of the file in the above fix (ST-8.5.1.0-Lotus-ST-IF-BMUY-8F9PUJ)</td>
</tr>
</tbody>
</table>
Install Fix Pack
To use the IBM Installation Manager to apply the Sametime 8.5.1.1 fix pack, follow these steps:

1. Launch the IBM Installation Manager.
2. Select File → Preferences from the menu.
3. Click Add Repository.
4. Browse to the repository.config file within the Sametime proxy hotfix directory and click OK.
5. From the main screen of the IBM Installation Manager select Update.
6. Make sure that IBM Sametime Proxy server fix 8.5.1.1 is selected and click Next to complete the installation.
Deploy Fixes
Included in the fix pack are three WebSphere Application fixes:
- 7.0.0.3-WS-WAS-IFPM18909
- 7.0.0.3-WS-WAS-IFPM20402
- 7.0.0.3-WS-WAS-MultiOS-IFPM19599

In the example environment, the Sametime 8.5.1 server is installed alongside the Sametime 8.5.2 Meeting server and uses the same IBM Sametime Server Platform. Because the Sametime Server platform is based on the version of WebSphere Application server that comes with Sametime 8.5.2, this already contains the WebSphere Application Server fixes included in the iFix.

If the Sametime 8.5.1 server had been installed on its own server and used the version of WebSphere Application server that came with Sametime 8.5.1, then the fixes would need to be applied using the following steps:

1. Extract the files from the `Sametime8.5.1.1_WebSphereiFixes.zip` file.
2. Run the IBM Update Installer (`WASInstallDir/UpdateInstaller/update.exe`)
3. When prompted browse to the directory where the fixes are stored

Note that there is no harm in trying to apply the fixes as they will not be installed if they are not required.

Deploy new SametimeProxy ear file
This section details the replacement of the SametimeProxy ear file with one that will support integration with Connections and Portal and the application of WebSphere Application server fixes (if they are required). Refer to the following document for more information.

The SametimeProxy.ear file that is included in ST-8.5.1.0-Lotus-ST-IF-BMU-Y-8F9PUJ supports the integration with Connections 3.0.1 but does not support the integration with Portal 7.0.0.1. Instead of using the ear file from this fix, a new SametimeProxy.ear file (available from IBM Sametime Support by referencing http://www-01.ibm.com/support/docview.wss?uid=swg21509230) is used to support integration with both Connections 3.0.1 and Portal 7.0.0.1.

1. From Integrate Solutions Console for the Sametime 8.5.1 Proxy server (not the Integrated Solutions Console for the main Sametime 8.5.2 cell) select Applications →Application Types →WebSphere enterprise applications.

2. Select SametimeProxy in the list and click Update.

3. For the Application update options select Replace the entire application and browse to location of replacement file.

4. Click Next.
5. Click **Next** and **Next** again.

6. On Map modules to servers section, select all modules, select the **STProxyCell** and click **Apply**.

7. Click **Next** and **Finish**.

8. When you get the message "Application SametimeProxy installed successfully", click **Save** to save the master configuration.

**Copy JARS**

Included in ST-8.5.1.0-Lotus-ST-IF-BMUY-8F9PUJ are two jar files. Replace the files in the following locations with the files included in the ifix. For more information refer to the following technote: https://www-304.ibm.com/support/docview.wss?rs=899&uid=swg21474355

- WASInstallDir/AppServer/profiles/STAppProfile/optionallibs/stproxy/stcommsrvrtk.jar
- WASInstallDir/AppServer/profiles/STAppProfile/optionallibs/stproxy/pooledcontainers.jar

**Change session ID**

In order to support integration with Portal 7.0.0.1 complete the following steps:

1. Change the session ID cookie from the default JSESSIONID as described in the Portal wiki:
   
   http://www-10.lotus.com/ldd/portalwiki.nsf/dx/Changing_the_session_cookie_ID_on_the_Lotus_Sametime_proxy_server_V7

2. Restart proxy server.
Manually register Sametime proxy server

In order to be able to manage the Sametime proxy server from the existing Sametime System Console, it is necessary to register the server. Note that, as this is a Sametime 8.5.1.x proxy server, it is not possible to federate the node into the same cell as the Sametime System Console.

For detailed information on manually registering a server after installation, refer to the following document:


These are the steps taken to manually register the Sametime proxy server in the example environment.

1. On the server where the Sametime 8.5.1.1 proxy server is installed, edit `console.properties` (found in `installDir/STPServerCell/console`) and add the password for the Sametime System Console. The other fields should already be completed.

1:  SSCHostName=ssc.demos.ibm.com
2:  SSCHTTPPort=9443
3:  SSCUserName=wasadmin
4:  SSCPassword=password
5:  LogLevel=FINEST
6:  SSCSSLEnabled=TRUE
7:  SSCHTTPSPort=9443

2. Edit `productConfig.properties` (found in `installDir/STPServerCell/console`), add the password for the Sametime System Console and also change the name of the deployment plan to be a unique name that will define the Sametime 8.5.1.1 proxy in the System Console: For the example environment `8511proxy` is the deployment plan name used for the example environment.

1:  ProductType=com.ibm.lotus.sametime.proxyserver
2:  OfferingVersion=8.5.1.1
3:  ID=20101010_1748
4:  InstallType=Cell
5:  NodeIP=stproxy.demos.ibm.com
6:  NodeHostName=stproxy.demos.ibm.com
7:  WASDMNode=STProxyNode
8:  WASPassword=password
9:  WASInstall=E:/IBM/WebSphere/AppServer
10:  ProxyInstallLocation=E:/IBM/WebSphere/STPServerCell
11:  WASHost=stproxy.demos.ibm.com
12:  WASCell=STProxyCell
13:  WASNode=STProxyNode
14:  WASAppProfile=STPAppProfile
15:  WASAppServerName=STProxyServer
16:  WASDMPROfile=STPDMgrProfile
17:  WASDMServerName=dmgr
18:  WASSNProfile=STPSNAppProfile
19:  STCommunityServerHost=sametime.demos.ibm.com
20:  STCommunityServerPort=1516
21:  SSCEnabled=FALSE
22:  WASSoapPort=8881
23:  WASDMSoapPort=8603
3. Open a command prompt and switch to `installDir/STPServerCell/console`.
4. Type `registerproduct.bat`.
5. The resulting output should end with **Registration done successfully -true**

```
WASDMHost=stproxy.demos.ibm.com
InstallType:Cell
STCommunityServerHost=stmeeting.demos.ibm.com
WASUserID=wasadmin
WASHost=stproxy.demos.ibm.com
WASProfile:STDMgrProfile
WASNode=stmeetingProxyNode
ModelP:stproxy.demos.ibm.com
WASSoapPort:8881
DBType:DB2
WASDMHost:stproxy.demos.ibm.com
WASInstall:E:/IBM/WebSphere/AppServer
ProxyInstallLocation:E:/IBM/WebSphere/STPServerCell
STPServerName:STProxyServer
Creating deployment record for host -stproxy.demos.ibm.com
Deployment record created successfully.
Updating deployment record ...
Deployment record updated successfully.
Product registration completed successfully.
Registration done successfully -true
```

6. From the Integrated Solutions Console for the Sametime System Console select **Sametime System Console → Sametime Guided Activities → Install Sametime Proxy Server**.

7. If the deployment plan for the Sametime Proxy Server is present with a status of **Installed/Registered**, then the installation and registration process has completed successfully.
Define the Sametime Meeting server for the Sametime Proxy server

Now that the Sametime Proxy server has been registered with the Sametime System Console it is possible to use the Sametime System Console to manage the Sametime 8.5.1.1 proxy server configuration including defining the Sametime Meeting server to be used by the server for instant meetings. See 4.4, “Websphere Portal installation” on page 73 where this is done for the 852 proxy server.

In the example environment, in **Sametime System Console → Sametime Servers → Sametime Proxy Servers → 8511proxy**, the Sametime Meeting server is configured as follows:

<table>
<thead>
<tr>
<th>Entry</th>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of Meeting server</td>
<td>Sametime Meeting Server</td>
<td></td>
</tr>
<tr>
<td>Host Name</td>
<td>meeting.demos.ibm.com</td>
<td></td>
</tr>
<tr>
<td>Port</td>
<td>9080</td>
<td></td>
</tr>
</tbody>
</table>

When the configuration is complete, restart all of the Sametime 8.5.1.1 Proxy servers (including deployment manager and node).

**Sametime integration within Portal, Connections, iNotes, and Quickr Domino**

For integration elements where Sametime features and functionality are surfaced within another product:

- Sametime integration within Portal: See 7.4, “Portal and Sametime integration” on page 123 on how to configure Portal to surface Sametime functionality.
- Sametime integration within Connections: See 8.1, “Connections and Sametime integration” on page 127 on how to configure Connections to surface Sametime functionality.
- Sametime integration within iNotes: See 9.1, “iNotes -- Sametime integration” on page 145 on how to configure iNotes to surface Sametime functionality.
- Sametime integration within Quickr Domino: See 10.1, “Lotus Quickr for Domino and Sametime integration” on page 153 on how to configure Quickr to surface Sametime functionality.

**11.2.2 Sametime as a consumer**

This section details the integration elements where Sametime surfaces the functionality of another product.

In order for the Connections business card to be surfaced within the Sametime 8.5.2 web client, the Sametime 8.5.2 proxy server must be upgraded to Sametime 8.5.2 Interim Feature release 1 (IFR1). Upgrading the Sametime 8.5.2 proxy server to IFR1 requires that the Sametime System Console and then the Sametime Community server be upgraded to these versions first.

Refer to the following document for detailed information:

http://www-10.lotus.com/ldd/stwiki.nsf/dx/Updating_Sametime_servers_to_Interim_Feature_Release_1_st852ifr1
Prepare Sametime to support Connections business card

Download the following part numbers to install the required Sametime components:

<table>
<thead>
<tr>
<th>Part numbers</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIY8ML</td>
<td>Sametime System Console 8.5.2 IFR1</td>
</tr>
<tr>
<td>CIY9ML</td>
<td>Sametime Community server 8.5.2 IFR1</td>
</tr>
<tr>
<td>CIY0ML</td>
<td>Sametime Proxy server 8.5.2 IFR1</td>
</tr>
</tbody>
</table>

Note: Sametime 8.5.2 Interim Feature Release 1 cannot be uninstalled. The server(s) should be backed up before starting the installation in case the previous installation needs to be restored.

Apply Sametime 8.5.2 IFR1

To apply Sametime 8.5.2 IFR1:

1. **Upgrading Sametime System Console server:**
   
   The following steps were carried out to upgrade the Sametime System Console in the example environment to Sametime 8.5.2 IFR 1
   
   a. Shut down all the Sametime servers on the machine where the installation is being performed, except for the System Console server (ie: for the example environment the Sametime Community server and Sametime 8.5.2 Proxy server are shut down)
   
   b. Run **update.bat** to launch the fix pack.
   
   c. Click **Update** on the Installation Manager screen.
   
   d. Click **Next** and the 8.5.2 IFR1 fix pack should be already selected.
   
   e. Click **Next, Next** again and then **Update** to start the installation.
   
   f. Restart all the Sametime System Console servers
   
   g. To verify the installation, check the Welcome page of the Integrated Solutions Console for the updated release number - 8.5.2.1.
2. Upgrading Sametime Community server:

The following steps were carried out to upgrade the Sametime Community server in the example environment to Sametime 8.5.2 IFR1

a. Make sure that the Sametime Community server is shut down.

b. Run `setupwin32` to launch the fix pack installation.

c. To verify the installation, check the Integrated Solutions Console (Sametime System Console → Sametime Servers → Sametime Community Servers) to confirm that the Sametime Community Server shows the updated release number 8.5.2 IFR 1.

3. Upgrading Sametime Proxy server:

The following steps were carried out to upgrade the Sametime 8.5.2 Proxy server in the example environment to Sametime 8.5.2 1IFR1

a. Leave the Sametime System Console WebSphere Application servers running and shut down the Sametime 8.5.2 Proxy server and node agent.

b. Run `update.bat` to launch the fix pack.

c. Click Update on the Installation Manager screen.

d. Click Next and the 8.5.2 IFR1 fix pack should be already selected.
e. Click **Next**, **Next** again and then **Update** to start the installation.

f. To verify the installation, check the Integrated Solutions Console (**Sametime System Console →Sametime Servers →Sametime Proxy Servers**) to confirm that the Sametime Proxy Server shows the updated release number 8.5.2 IFR 1.

**Note:** The Sametime proxy server will have been restarted as part of the upgrade process.

### 11.2.3 Connections integration within Sametime

See 11.1, “Overview of Sametime integration” on page 161 on how to configure Sametime to surface Connections functionality.
Creating common navigation across the products

This section describes how to customize the products in the integrated environment to make them look and act like a single product by:

- Providing navigation functionality that allows users to move easily between the products in the environment.
- Using a common look and feel, branding and navigation within as many products as possible or appropriate.

This section includes the following subsections:

- 12.1, “Customizing the WebSphere Portal theme” on page 176
- 12.2, “Customizing IBM Connections theme” on page 206
- 12.3, “Customization of Lotus Quickr Domino theme” on page 209
- 12.4, “Navigating from iNotes to other products” on page 222
12.1 Customizing the WebSphere Portal theme

This section describes how to customize Portal to include a logo for the integrated environment and provide navigation options in the Portal banner that give access to Connections and Quickr Domino.

12.1.1 Overview

In this section we will customize the WebSphere Portal v7 banner to match that of IBM Connections. In order to do this we will have to backup directories containing the old theme, alter existing files and add some new files.

The default Portal banner looks like this:

...with the following additions and changes:

1. A logo is added (which is also added to the Quickr Domino and Connections banners).
2. Menu option for Home is changed to Intranet - Welcome to reflect the name used on the Quickr Domino and Connections banners for accessing Portal.
3. Menu options are added for direct access to Connections Profiles and Communities.
4. Menu option is added for direct access to Quickr Domino.
5. Drop down menu is added for direct access to other Connections applications.
6. Other default menu options are removed to keep the banner as clear and simple as possible and the search bar is also moved out of the banner (into the title bar).

DISCLAIMER: IBM DOES NOT WARRANT OR REPRESENT THAT THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE PROVIDED ARE COMPLETE OR UP-TO-DATE. IBM DOES NOT WARRANT, REPRESENT OR IMPLY RELIABILITY, SERVICEABILITY OR FUNCTION OF THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE. IBM IS UNDER NO OBLIGATION TO UPDATE CONTENT NOR PROVIDE FURTHER SUPPORT.

ALL THE EXAMPLE INTEGRATED WEB ENVIRONMENT SETUP AND THE CODE ARE PROVIDED "AS IS," WITH NO WARRANTIES OR GUARANTEES WHATSOEVER. IBM EXPRESSLY DISCLAIMS TO THE FULLEST EXTENT PERMITTED BY LAW ALL EXPRESS, IMPLIED, STATUTORY AND OTHER WARRANTIES, GUARANTEES, OR REPRESENTATIONS, INCLUDING, WITHOUT LIMITATION, THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF PROPRIETARY AND INTELLECTUAL PROPERTY RIGHTS. YOU UNDERSTAND AND AGREE THAT YOU USE THESE MATERIALS, INFORMATION, PRODUCTS, SOFTWARE, PROGRAMS, AND SERVICES, AT YOUR OWN DISCRETION.
12.1.2 Changes to files within the Portal server PageBuilder2 directory

In this section we will alter existing files, create a directory for custom files and create these custom files.

**Backup directory to be changed**

The content of the following directory will be changed as a result of the following customization activities so a backup should be taken before making any changes or additions:

- `WebSpherePortalInstallDirectory\PortalServer\theme\wp.mashup.cc.theme\installedApps\wp.mashup.cc.theme.ear\PageBuilder2.war\themes\html\PageBuilder2`

**Create subdirectory in PageBuilder2 directory to hold the custom files**

Create a subdirectory in the PageBuilder2 directory where the custom files are stored. In the example environment this subdirectory is called `redwiki_cust`.

**Create bannerNavConfig.jspf**

In the `redwiki_cust` subdirectory, create a file `bannerNavConfig.jspf` to act as the host configuration file. If the host name of the Connections or Quickr Domino server changes, only this file needs to be edited.

Add the following code to the file:

```java
1: <
2: String quickrServerHostname = "quickr.demos.ibm.com";
3: String connectionsServerHostname = "connections.demos.ibm.com";
4: %>
```

**Create apps.jsp**

In the `redwiki_cust` subdirectory create a file `apps.jsp` to contain the code for the "Apps" drop down menu that provides links to the Connections components.
Add the following code to this file:

```
<%@ page contentType="text/html;charset=UTF-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>

<%@ include file="./bannerNavConfig.jspf" %>

<div role="document">
<table cellspacing="0" cellpadding="0" class="lotusLayout lotusNavMenuLarge">
<tbody>
<tr>
<th class="lotusNowrap" scope="row">
<img role="presentation" alt="" src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" class="lconnSprite lconnSprite-iconActivities16">
<a href="//<%=connectionsServerHostname%>/activities"><strong>Activities</strong></a>
</th>
<td class="lotusNowrap">
<a href="//<%=connectionsServerHostname%>/activities/service/html/mainpage#todolist">To Do List</a>
</td>
<td class="lotusNowrap lotusLastCell">
<a href="//<%=connectionsServerHostname%>/activities/service/html/mainpage#dashboard, highpriority">High Priority Activities</a>
</td>
</tr>
<tr>
<th class="lotusNowrap" scope="row">
<img role="presentation" alt="" src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" class="lconnSprite lconnSprite-iconBlogs16">
<a href="//<%=connectionsServerHostname%>/blogs"><strong>Blogs</strong></a>
</th>
```
Create communities.jsp
In the redwiki_cust subdirectory create a file communities.jsp to contain the code for the "Communities" drop down menu.

Add the following code to the file:

```jsp
<%@ page contentType="text/html;charset=UTF-8" %>
<%@ taglib prefix="c"       uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="fn"     uri="http://java.sun.com/jsp/jstl/functions" %>
<%@ taglib prefix="fmt"    uri="http://java.sun.com/jsp/jstl/fmt" %>
<c:if test="${param.authenticated == 'true'}" var="isAuthenticated" scope="page" />
<%@ include file="./bannerNavConfig.jspf" %>

<div role="document">
<table dojoType="dijit._Widget" class="lotusLayout" cellpadding="0" cellspacing="0">

<--
When the user is authenticated, we will call the Community REST API to retrieve the seven most recently updated communities the user is following. The menu will wait a fraction of a second after opening before retrieving the list of communities, so that moving the mouse cursor over the header menus rapidly will not begin the request.
-->

<c:if test="${isAuthenticated}">
<script type="dojo/method" event="postCreate">
    this.parent = dijit.getEnclosingWidget(this.domNode.parentNode);
    this.connect(this.parent, "onOpen", "onParentOpened");
    dojo.query("[dojoAttachPoint]", this.domNode).forEach(function(n) {
        this[dojo.attr(n, "dojoAttachPoint")] = n;}, this);
    this.delayLoad();
</script>

<script type="dojo/method" event="onParentOpened">
    this.delayLoad();
</script>

<script type="dojo/method" event="expire">
    this.loaded = false;
</script>

<script type="dojo/method" event="delayLoad">
    setTimeout(dojo.hitch(this, "load"), 325);
</script>

<script type="dojo/method" event="load">
    if (!this.parent._launcher._opened)
        return;
    if (this.isLoading || this.loaded)
        return;
    this.isLoading = true;
    var url = "http://<%=connectionsServerHostname%>/communities/service/atom/forms/communities/my?sortField=lastmod&sortOrder=descending&filterType=following&ps=7";
    var dfd = dojo.xhrGet({url: url, handleAs: "xml", auth: {secured: false}});
    dfd.addCallback(this, "loadCommunities").addErrback(this, "loadError");
</script>

<script type="dojo/method" event="loadCommunities" args="doc">
    this.loaded = true; this.isLoading = false;
    setTimeout(dojo.hitch(this, "expire"), 10*60*1000);
    var feed = doc.documentElement;
    var MAX_NAME_LENGTH = 50;
var entries = feed.getElementsByTagName("entry");

var t = this.template;
var tbody = t.parentNode;
while (t.nextSibling)
    tbody.removeChild(t.nextSibling);
this.empty.style.display = this.info.style.display =
this.loading.style.display = this.error.style.display = "none";

if (entries.length > 0) {
    var items = [];
    for (var i=0,c=Math.min(entries.length,7); i<c; i++) {
        var entry = entries[i];
        var title = entry.getElementsByTagName("title")[0];
        title = title.text || title.textContent;
        items.push({
            title: title,
            link: dojo.query("link[rel='alternate'][type='text/html']",
entry)[0].getAttribute("href")
        });
    }
    items.sort(function(a,b) {if (a.title > b.title) return 1; if
(a.title < b.title) return -1; return 0;});
    this.info.style.display = "";
    for (var i=0,c=Math.min(items.length,7); i<c; i++) {
        var item = items[i];
        var title = item.title;
        var shortTitle = title.length > MAX_NAME_LENGTH ?
(dojo.trim(title.substring(0,MAX_NAME_LENGTH)) + "...") : title;
        var link = item.link;
        var tr = t.cloneNode(true); tr.style.display = ""; dojo.attr(tr,
"dojoAttachPoint", null);
        var td = tr.firstChild;
        var a = td.appendChild(document.createElement("a")); a.className =
"lotusCommunity"; a.href = link;
        if (title != shortTitle)
            a.title = title;
        a.appendChild(document.createTextNode(shortTitle));
        tbody.appendChild(tr);
    }
} else
    this.empty.style.display = "";
console.log("load done");
</script>
<script type="dojo/method" event="loadError" args="error">
    this.isLoading = false;
    var t = this.template;
    while (t.nextSibling)
        t.parentNode.removeChild(t.nextSibling);
    this.loading.style.display = this.info.style.display =
    this.empty.style.display = "none";
    this.error.style.display = "";
    this.error.title = error ? (error.message || error.code || error) :
null;
These are the normal menu items displayed for all users.

These menu items are only visible to authenticated users.
Create people.jsp

In the redwiki_cust subdirectory create a file people.jsp to contain the code for the "Profiles" drop down menu.

Add the following code to the file.

```
<%@ page session="false" buffer="none" %>
<%@ page trimDirectiveWhitespaces="true" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/json" prefix="json" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/portal-fmt" prefix="portal-fmt" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/portal-core" prefix="portal-core" %>
```
Create places.jsp
In the redwiki_cust subdirectory create a file places.jsp to contain the code for the “Project Places” drop down menu.
Add the following code to the file:

```html
<%@ page session="false" buffer="none" %>
<%@ page trimDirectiveWhitespaces="true" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/json" prefix="json" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/portal-fmt" prefix="portal-fmt" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/portal-core" prefix="portal-core" %>
<%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/resolver" prefix="r" %>
<%@ include file="./bannerNavConfig.jspf" %>

<div role="document" style="background-color:#FFFFFF">
<table dojoType="dijit._Widget" class="lotusLayout" cellpadding="0" cellspacing="0">
<tbody>
<tr>
<td class="lotusNowrap lotusLastCell">
<a id="myplaceslink" href="//<%=quickrServerHostname%>/places/myplaces"><strong id="myplacestext">My Places</strong></a>
</td>
</tr>
<tr>
<td class="lotusNowrap lotusLastCell">
<a id="publicplaceslink" href="//<%=quickrServerHostname%>/places/publicplaces"><strong id="publicplacestext">Public Places</strong></a>
</td>
</tr>
</tbody>
</table>
</div>
```
Create header.js

In the redwiki_cust subdirectory create a file `header.js` to contain the JavaScript code needed for the initialising the dojo widgets within the navigation bar.

Add the following code to the file:

```javascript
1: dojo.provide("lconn.core.header");
2: dojo.require("lconn.core.widget.MenuLauncher");
3: /
4: * We defer initializing the dojo widgets for the menu until a menu is focused or hovered. These methods assume
5: * that all parameters needed to declare a header menu are statically defined on the link. The only required attribute
6: * is 'src' which is the URL to access to load the contents of the menu (see dijit.Dialog and the href parameter for
7: * more information on how header menus are loaded).
8: */
9: lconn.core.header = {
10:    initMenu: function(a,type) {
11:        a._init = true;
12:        var launcher = new lconn.core.header.MenuLauncher({
13:            menuHref: dojo.attr(a, "src"),
14:            menuId: dojo.attr(a, "aria-owns"),
15:            dialogTitle: dojo.attr(a, "aria-label"),
16:            focusNode: a
17:        }, a.parentNode);
18:        launcher[type]();
19:    },
20:    menuFocus: function(a) {
21:        if (a._init)
22:            this.initMenu(a,"onFocus");
23:    },
24:    menuClick: function(a) {
25:        if (a._init)
26:            this.initMenu(a,"onClick");
27:    },
28:    menuMouseover: function(a) {
29:        if (a._init)
30:            this.initMenu(a,"onMouseEnter");
31:    },
32:    enableLanguageSelector: function(node, languages, cookieName, cookieDomain, cookieTimeout) {
33:        // use dj.require to ensure that LanguageSelector and lconn.core.url
34:        var dj = dojo;
35:    }
```
`var properties = {path: "/"};

var timeout = cookieTimeout == -1 ? 0 : (cookieTimeout / 86400);
if (timeout > 0)
    properties.expires = timeout;

var domain = cookieDomain;
if (domain && domain != ".") {
    dj.require("lconn.core.url");
    var uri = lconn.core.url.parse(window.location.href);
    var host = uri.host;
    properties.domain = domain + host.substring(host.indexOf('.'), host.length);
}

var node = dojo.byId(node);
if (!node) {
    console.log("language selector node not present in header or footer");
    return;
}

try {
    dj.require("lconn.core.LanguageSelector");
    var next = node.nextSibling;
    while (next) {
        if (next.nodeType == 1)
            dojo.removeClass(next, "lotusFirst");
        next = next.nextSibling;
    }
    node.innerHTML = "<a id='headerLanguageSelectorMenu' href='javascript:' role='button' aria-haspopup='true' aria-owns='headerLanguageSelectorMenu_popup'></a>";
    node.style.display = "";
    new lconn.core.LanguageSelector(node.firstChild, languages, cookieName, properties);
}

catch (e) {
    console.error("lconn/core/LanguageSelector.js unavailable ");
    console.error(e);
}

switchTheme: function(themeId) {
    var baseLink = dojo.byId("lotusBaseStylesheet");
    var themeLink = dojo.byId("lotusThemeStylesheet");
    if (themeLink) {
        themeLink.id = "";
        setTimeout(function() {if (themeLink.parentNode)
            themeLink.parentNode.removeChild(themeLink);}, 1);
    }
    if (themeId) {
        var newLink = baseLink.cloneNode(true);
        newLink.id = "lotusThemeStylesheet";
var m = /
 ^common\//styles\//base\//base\.css\([^\\w]+\)?\.exec(baseLink.href);

var uriTheme = m[1] + 
"/common/styles/"+encodeURIComponent(themeId)+"Theme/theme.css" + (m[2] || 
"");

newLink.href = uriTheme;
dojo.place(newLink, baseLink, "after");

/**
 * This class will dynamically define the menu class. Some components defer
 loading the dialog function because of size, so this method
 * will not statically include dijit.Dialog. Instead, components should
 either define dijit.Dialog as an explicit dependency in their dojo
 * build scripts (if not already required by another class) or override
 lconn.core.header.MenuLauncher.prototype._requireDialog
 */
dojo.declare("lconn.core.header.MenuLauncher",
lconn.core.widget.MenuLauncher, {
    activeParent: 0, // the parent node will have the "lotusHover" class
    _initMenuFinal: function() {
        if (!dojo.getObject("lconn.core.header.Menu")) {
            var d = dojo;
d.provide("lconn.core.header.Menu");
dojo.declare("lconn.core.header.Menu", dijit.TooltipDialog, {
        autofocus: false,
        "class": "lotusNavMenu",
        postCreate: function() {
            this.inherited(arguments);
            dijit.setWaiState(this.containerNode, "label",
            this.dialogTitle);
    },
    _attachTemplateNodes: function() {
        this.inherited(arguments);
    },
    // connect the hover events so that hide/show can also apply
    when the user mouses over the popup
    connect(this.domNode, "onmouseover", "onMouseOver");
    connect(this.domNode, "onmouseout", "onMouseOut");
    connect(this.domNode, "onclick", "onMenuClick");

    orient: function(/*DomNode*/ node, /*String*/ aroundCorner,
    /*String*/ corner){
        // summary: configure widget to be displayed in given
    position relative to the button
    this.domNode.className = this["class"] +
    dijitTooltipAB+(corner.charAt(1)=='L'?'Left':"Right");+/
    dijitTooltip"+(corner.charAt(0)=='T' ? "Below" : "Above");

    },
    onMenuClick: function(e) {
        var el = e.target;
    for (var i=0; el && i<5; i++)
    if (el.nodeName.toLowerCase() == "a") {
        // Internet Explorer is not properly clearing the
: hover state when the link is clicked. To ensure
// reset of the state, we set the visibility here. The
// setTimeout was necessary
// to force the reflow. Doing measurement style reflow
did not reset :hover. Also, note that
// no mouseout is fired on the element when the click
hides the menu div - this is probably
// related to the underlying issue.
if (dojo.isIE < 9) {
    var style = el.style;
    style.visibility = "hidden";
    setTimeout(function() {style.visibility=""},1);
}
this.onCancel();
return;
else
    el = el.parentNode;
}
}
this.menu = new lconn.core.header.Menu({
    widthAdjust: 8,
    href: this.menuHref,
    dialogTitle: this.dialogTitle,
    id: this.menuId || undefined
});
/**
* Provided so that some callers can load in a deferred manner.
*/
_initMenu: function() {
    return this._whenDialog().addCallback(this, "_initMenuFinal");
},
_whenDialog: function() {
    var dfd = this._dlgDfd;
    if (!dfd) {
        var d = dojo;
        d.require("dijit.Dialog"); // Use the indirect load so that the
static build scripts do not include dijit.Dialog automatically.
        dfd = this._dlgDfd = new dojo.Deferred(); dfd.callback();
    }
    return dfd;
};

Create MenuLauncher.js
In the redwiki_cust subdirectory create a file MenuLauncher.js to contain the JavaScript
code used to launch the drop down menu items.

Add the following code to the file:

```javascript
/**
*/
```
* Enable a link that will launch a menu. This class also provides the capability to
* instantiate the menu on demand. You must pass an existing link as the source ref.

```
dojo.provide("lconn.core.widget.MenuLauncher");
dojo.require("dijit._Widget");
dojo.declare("lconn.core.widget.MenuLauncher", dijit._Widget, {
  hideDelay: 50,
  openDelay: 0,
  preloadDelay: 0, // if greater than zero, load in the background during the open delay

  classActive: "lotusHover",
  activeParent: 0,
  /** Set to true if you want focus events to open this menu */
  activateOnFocus: false,

  widthAdjust: 0,

  orient: {'BL':'TL', 'BR':'TR', 'TL':'BL', 'TR':'BR'},
  orientRTL: {'BR':'TR', 'BL':'TL', 'TR':'BR', 'TL':'BL'},

  //optMenu: {}, //arguments that should be passed to the menu when initialized

  buildRendering: function() {
    this.domNode = this.popupStateNode = this.srcNodeRef;
    this.connect(this, "onMouseEnter", "_delayOpen");
    this.connect(this, "onMouseLeave", "_delayClose");
    if (this.activateOnFocus) {
      this.connect(this, "onFocus", "_delayOpen");
      this.connect(this, "onBlur", "_closeMenu");
    } else
      this.connect(this, "onClick", "_onClick");
    this.menuId = this.menuId || dojo.attr(this.domNode, "aria-owns") ||
    dojo.attr(this.domNode, "aria-describedby") || (this.id ? (this.id +"_popup") :
    dijit.getUniqueId("ml"));
    this.decorateAria();

    // We're done with the srcNode at this point
    // Removing it fixes an IE focus-loss problem replacing a DOM node with itself
    delete this.srcNodeRef;
  },

  decorateAria: function() {
    dojo.attr(this.domNode, "aria-owns", this.menuId);
  },

  whenInit: function() {
    var dfd = this._initDfd;
    if (!dfd) {
```
var ret = this._initMenu();
if (!ret instanceof dojo.Deferred) {
    if (!this.menu)
        throw "_initMenu must initialize this.menu or return a deferred";
    ret = new dojo.Deferred();
    ret.callback(this.menu);
} 
dfd = this._initDfd = ret;
dfd.addCallback(this, "initFinal").addErrback(this, function()
    {this._initDfd = null;});
return dfd;

/**
 * Called after this.menu is set
 */
initFinal: function() {
    var el = this.domNode;
    // attach a hidden widget reference node to ensure that the widget gets destroyed
    var menu = this.menu;
    menu._launcher = this;
    dijit.popup.moveOffScreen(menu.domNode);
    //dijit.popup.prepare(menu.domNode);
    var span = document.createElement("span");
    span.style.display = "none";
    dojo.attr(span, "widgetId", menu.id);
    el.parentNode.appendChild(span);

    var activeNode = this.domNode;
    for (var i=0; i<this.activeParent; i++)
        activeNode = activeNode.parentNode;
    this.activeNode = activeNode;

    this.connect(menu, "onMouseEnter", "_openMenu");
    this.connect(menu, "onMouseLeave", "_delayClose");
},

/**
 * Should create a dropdown widget and set it to this.menu. May optionally
 * return a dojo.Deferred to indicate that processing may take some time.
 */
_initMenu: function() {
    throw "Must override initMenu() to do dynamic loading";
},

_onClick: function/*Event*/ e) {
    if (e) dojo.stopEvent(e);
    this._openMenu("click");
},

_preloadMenu: function() {
    this.whenInit().addCallback(this, function() {

var menu = this.menu;
var dlcp = dojo.getObject("dijit.layout.ContentPane");
if (menu._preload)
    menu._preload();
else if (dlcp && menu instanceof dlcp) {
    // only preload if there are no inflight requests
    if (menu.isLoaded || menu._xhrDfd)
        return;
    menu._load();
    menu.refresh();
    }
},
_openMenu: function(eventType) {
    this._shouldBeOpen = true;
    this.whenInit().addCallback(this, function() {
        if (!this._opened && this._closing) || !this._shouldBeOpen
            //_delayClose was called before init completed, and item has not
            yet been opened
            return;
        
        if (typeof eventType != "string")
            eventType = null;
        this._delayEnd();
        var menu = this.menu;
        if (!menu) { return; }
        if (!this._opened) {
            // If there's an href, then load that first, so we don't get a
            flicker
            if (menu.href && !menu.isLoaded) {
                var self = this;
                var handler = dojo.connect(menu, "onLoad", function()
                    { doj
                        disconnect(handler);
                        self._openFinal(eventType);
                    },
                );
                this._preloadMenu();
                return;
            }
        }
        else
            this._openFinal(eventType);
    }),
    this._shouldBeOpen = true;
    this.whenInit().addCallback(this, function() {
        if (!this._opened && this._closing) || !this._shouldBeOpen
            //_delayClose was called before init completed, and item has not
            yet been opened
            return;
        
        if (typeof eventType != "string")
            eventType = null;
        this._delayEnd();
        var menu = this.menu;
        var oldWidth=menu.domNode.style.width;
        var self = this;
if (lconn.core.widget.MenuLauncher._menu)
    dijit.popup.close(lconn.core.widget.MenuLauncher._menu);

if (this.onBeforeOpen(eventType) === true)
    return;

dojo.addClass(this.activeNode, this.classActive);

dijit.popup.open({
    parent: this,
    popup: menu,
    around: this.domNode,
    orient: this.getOrient(),
    onExecute: function(){
        self._closeMenu(true);
    },
    onCancel: function(){
        self._closeMenu(true);
    },
    onClose: function(){
        self._delayEnd();
        lconn.core.widget.MenuLauncher._menu = null;
        menu.domNode.style.width = oldWidth;
        dijit.setWaiState(menu.domNode, "hidden", "true");
        if (self.domNode)
            dojo.removeClass(self.activeNode, self.classActive);
        self._opened = false;
        self.onClose();
    }
});

lconn.core.widget.MenuLauncher._menu = menu;

this._opened=true;

this.onOpen(eventType);

__closeMenu: function(/*Boolean*/ focus){
    this._delayEnd();
    this._shouldBeOpen = false;
    if (this._opened) {
        if (dojo.isIE)
            setTimeout(dojo.hitch(this, "__closeMenuFinal", focus),0);
        else
            this._closeMenuFinal(focus);
    }
};

__closeMenuFinal: function(/*Boolean*/ focus){
    this._shouldBeOpen = false;
    dijit.popup.close(this.menu);
    if (focus) { dijit.focus(this.focusNode || this.domNode); }
};

getOrient: function() {
return this.isLeftToRight() ? this.orient : this.orientRTL,
 },

function() {
   if (!this._opening) {
      this._delayEnd();
      this._opening = setTimeout(dojo.hitch(this, "_openMenu", "mouseover"), this.openDelay);
      if (this.preloadDelay > 0 & !this.menu || !this.menu.isLoaded))
         this._preload = setTimeout(dojo.hitch(this, "_preloadMenu"), this.preloadDelay);
   }
}

function() {
   if (!this._closing) {
      this._delayEnd();
      this._closing = setTimeout(dojo.hitch(this, "_closeMenu", false), this.hideDelay);
   }
}

function() {
   try {clearTimeout(this._opening);} catch (e) {}
   try {clearTimeout(this._closing);} catch (e) {}
   try {clearTimeout(this._preload);} catch (e) {}
   this._closing = this._opening = this._preload = null;
}

/*
* If true is returned the open will be aborted
*/

function(eventType) {
   return false;
}

/*
* Type will be null if the event that launched this menu was a hover
*/

function(eventType) {
   if (eventType == "click" && !this.activateOnFocus)
      this.focusMenu();
   dijit.setWaiState(this.menu.domNode, "hidden", "false");
}

function() {
}


focusMenu: function() {
   var menu = this.menu;
   if (menu.focus)
      menu.focus();
   else {
      menu._getFocusItems(menu.containerNode);
      dijit.focus(menu._firstFocusItem);
   }
}
Create images directory
In the redwiki_cust subdirectory create an images subdirectory to contain the necessary custom images.

Edit BannerNav.jsp
BannerNav.jsp (stored in WebSpherePortalInstallDirectory\PortalServer\theme\wp.mashup.cc.theme\installedApps\wp. mashup.cc.theme.ear\PageBuilder2.war\themes\html\PageBuilder2) contains the code for the top navigation bar, we edit this to contain our new options and point to the custom files which control these. Text in bold indicates the additions to make to this file.

1: 2: 3:  <%@ page session="false" buffer="none" %>
4:  <%@ page trimDirectiveWhitespaces="true" %>
5:  <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
6:  <%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/json" prefix="json" %>
7:  <%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/portal-fmt" prefix="portal-fmt" %>
8:  <%@ taglib uri="http://www.ibm.com/xmlns/prod/websphere/portal/v7.0/portal-core" prefix="portal-core" %>
9:  10:  <%@ include file="./redwiki_cust/bannerNavConfig.jspf" %>
15:  16:  <div id="topNavLinksContainer">-- Renders the links in the banner (Home, Admin, etc.) server side --</div>
17:  18:  <c:set var="node" value="${wp.selectionModel.selectionPath[0]}"/><c:set var="nodeID" value="${wp.identification[node]}"/>
19:  20:  <div id="topNavLinks" class="" data-nm-level="0" data-nm-primed="" if test=""${primeNavigation}""><portal-fmt:out><portal-core:navigationNodePriming navigationNode=""${nodeID}"" metaData=""${navHiddenMetadata}"" considerChildren="false" /></portal-fmt:out></c:if>
21:  22:  <ul class="lotusInlinelist lotusLinks" role="navigation">
23:  24:  <li id="${nodeID}"
25:  26:  <c:choose>


27:    <c:when test="${node.metadata['com.ibm.portal.Hidden']}">
28:      ' style="display:none"'
29:    </c:when>
30:    <c:when test="${node.metadata['showcaseHiddenPage']}">
31:      ' style="display:none"'
32:    </c:when>
33:    <c:when test="${node.title eq 'Administration'}">
34:      ' style="display:none"'
35:    </c:when>
36:     <c:otherwise>
37:      ' 
38:    </c:otherwise>
39:  </c:choose>
40:
41:  class="lotusTab modelHasChildren<c:if test="${wp.selectionModel[node] != null}"> lotusSelected</c:if>"
42:  data-mm-primed="<c:if test="${primeNavigation}">
43:  <portal-fmt:out><portal-core:navigationNodePriming
44:  navigationNode="${nodeID}" metaData="${navHiddenMetadata}"
45:  considerChildren="false" /></portal-fmt:out>
46:  </c:if>">
47:  <a href="?uri=nm:oid:${nodeID}"
48:  <c:choose>
50:  <c:otherwise>
51:  <portal-fmt:out>${node.title}</portal-fmt:out>
52:  </c:otherwise>
53:  </c:choose>
54:  </a>
55:  </li>
56:  </ul>
57:  </div>
58:  </div>
59:
60:
61:  <ul class="lotusInlinelist lotusLinks" role="navigation">
62:    <li id="lotusBannerProfiles" aria-owns="lconnheadermenu-people" widgetid="lotusBannerProfiles">
63:      <a href="http://<%=connectionsServerHostname%>/profiles"
64:     src="/PageBuilder2/themes/html/PageBuilder2/redwiki_cust/people.jsp"
65:     aria-label="Profiles" aria-owns="lconnheadermenu-people" role="button"
66:    onfocus="dojo.require('lconn.core.header');lconn.core.header.menuFocus(this);
67:    onclick="dojo.require('lconn.core.header');lconn.core.header.menuClick(this);
68:    onmouseover="dojo.require('lconn.core.header');lconn.core.header.menuMouseover(this)
69:    ">
70:    Profiles <img class="lotusArrow lotusDropDownSprite"
71:    src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" alt="" role="presentation">
72:    </a>
73:  </li>
74:  </ul>
<li class="" id="lotusBannerCommunities"
aria-owns="lconnheadermenu-communities" widgetid="lotusBannerCommunities">
<a href="http://<%=connectionsServerHostname%>/communities" src="/PageBuilder2/themes/html/PageBuilder2/redwiki_cust/communities.jsp?authenticated=false" aria-label="Communities" aria-owns="lconnheadermenu-communities" role="button"
onfocus="dojo.require('lconn.core.header');lconn.core.header.menuFocus(this);"
onclick="dojo.require('lconn.core.header');lconn.core.header.menuClick(this);"
onmouseover="dojo.require('lconn.core.header');lconn.core.header.menuMouseover(this);" id="lotusBannerCommunitiesLink">
Communities <img class="lotusArrow lotusDropDownSprite" src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" alt="" role="presentation">
</a>
</li>

<li id="lotusBannerPlaces" class="">
<a onmouseover="dojo.require('lconn.core.header');lconn.core.header.menuMouseover(this);"
onclick="dojo.require('lconn.core.header');lconn.core.header.menuClick(this);"
onfocus="dojo.require('lconn.core.header');lconn.core.header.menuFocus(this);" role="button" aria-owns="lconnheadermenu-places" aria-label="Places" src="/PageBuilder2/themes/html/PageBuilder2/redwiki_cust/places.jsp" href="http://<%=quickrServerHostname%>/places" />
Project Places <img role="presentation" alt="" src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" class="lotusArrow lotusDropDownSprite">
</a>
</li>

<li class="" id="lotusBannerApps" aria-owns="lconnheadermenu-apps"
widgetid="lotusBannerApps">
<a href="javascript:" src="/PageBuilder2/themes/html/PageBuilder2/redwiki_cust/apps.jsp" aria-label="Apps" aria-owns="lconnheadermenu-apps" role="button"
onfocus="dojo.require('lconn.core.header');lconn.core.header.menuFocus(this);"
onclick="dojo.require('lconn.core.header');lconn.core.header.menuClick(this);"
onmouseover="dojo.require('lconn.core.header');lconn.core.header.menuMouseover(this);">
Apps <img class="lotusArrow lotusDropDownSprite" src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" alt="" role="presentation">
</a>
</li>

<%-- This is a quick way to get the 'Administration' top page to appear at the end of the global banner --%>
<%-- It might not be the most efficient way. This is used for demonstration purpose only. --%>
Chapter 12. Creating common navigation across the products

97: <div id="topNavLinksContainer">
98: <c:set var="node" value="${wp.selectionModel.selectionPath[0]}"/>
99: <c:set var="nodeID" value="${wp.identification[node]}"/>
100: <ul class="lotusInlinelist lotusLinks" role="navigation">
101: <c:forEach var="node" items="${wp.navigationModel.children[node]}">
102: <li id="${nodeID}-two"
103: <c:choose>
104: <c:when test="${node.title eq 'Administration'}">
105: ' ' 
106: </c:when>
107: <c:otherwise>
108: ' style="display:none"'
109: </c:otherwise>
110: </c:choose>
111: class="lotusTab modelHasChildren<c:if test="${wp.selectionModel[node] != null}"> lotusSelected</c:if>"
112: data-nm-primed="<c:if test="${primeNavigation}">
113: <portal-fmt:out><portal-core:navigationNodePriming navigationNode="${nodeID}" metaData="${navHiddenMetadata}" considerChildren="false" /></portal-fmt:out></c:if>">
114: <a href="?uri=nm:oid:${nodeID}"
115: <c:choose>
118: <c:otherwise>
119: <portal-fmt:out>${node.title}</portal-fmt:out>
120: </c:otherwise>
121: </c:choose>
122: </a>
123: </li>
124: </c:forEach>
125: </ul>
126: </div>
127: </div>
128: <%-- This iWidget definition will replace the server side code above with an instance of the navigation widget --%>
129: <%--
130: <div class="iw-iWidget iw-Standalone" id="bannerNav"
131: <a class="iw-Definition" href="${ccConfig['cc.builder.context']}/widget-catalog/ControlledNavWidget.xml?pragma=cache&max-age=1209600&cache-scope=public&vary=none&user-context=false">
132: <span class="iw-ItemSet" style="display:none" title="attributes">
133: <a class="iw-Item" href="#controller">ibmCfg.controllers.navigation</a>
134: </span>
135: </a>
12.1.3 PageBuilder2 changes within WebDAV directory

The rest of the changes are done using a WebDAV client to access files in the Portal database.

These changes include:

- Moving the search bar off the banner onto the title page
- Adding and positioning the logo on the banner

Set up WebDAV client and connecting to Portal server

A webDAV client is used to connect to the Portal server. In this example a third party tool called WebDrive (http://www.webdrive.com/products/webdrive/) is used. This is installed on the Portal server and used to access the file store (fs-type1) from where static resources can be managed.

1. Launch WebDrive, select **New Site** and use the following values to configure a connection.

<table>
<thead>
<tr>
<th>Entry</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of server</td>
<td>Connect by Server Type (SFTP, WebDAV, FTP etc)</td>
</tr>
<tr>
<td>Server type</td>
<td>WebDAV/</td>
</tr>
<tr>
<td>URL for server</td>
<td><a href="http://localhost:10039/wps/mycontenthandler/fs-type1">http://localhost:10039/wps/mycontenthandler/fs-type1</a></td>
</tr>
<tr>
<td>Username</td>
<td>Administration credentials for the Portal server</td>
</tr>
<tr>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>Site name</td>
<td>portal fs-type 1</td>
</tr>
<tr>
<td>Drive letter</td>
<td>Z:</td>
</tr>
</tbody>
</table>

2. Check **Connect to Site now** and click **Finish**. Note: It may take up to 20 seconds to connect, but when it does, a window will display the contents of the fs-type1 entry point.
Backup directories to be changed
The content of the following directory will be changed during the customization so a backup of this should be taken at this stage:

- `WebDAVDrive\themes\PageBuilder2`

Adding a logo to the banner
Place the logo that you want to use in the css/images directory.

Edit `nls/theme_en.html`
This file is the Portal global theme file for English. If using another language, the changes within this file will need to be transferred to the corresponding file for that language.

The changes made here move the Portal search bar out of the banner and into the title bar.

```html
1: <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
2: <head>
3: <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
4: <!-- rel=dynamic-content indicates an element that is replaced with the contents produced by the specified href. -->
5: <link rel="dynamic-content" href="dyn-cs:id:head@tl:oid:csa2.theme">```
```html
<script type="text/javascript">
dojo.addOnLoad(function(){
  com.ibm.pb.themes.commonInit({
    setWindowTitle:false, // title is set by the head dynamic content spot
    useNavigationController: true,
    useRenderingController: true,
    useDNDController: true,
    initLiveTextService: true,
    lazyLoadModeWidgets: true,
    navPrimingContainers: ["selectionPathPrimer","topNavLinks","navTabsRoot"],
    customInit: com.ibm.themes.PageBuilder2.init
  });
dojo.publish("com.ibm.portal.theme.portlet_ready"); // notifies ASA that portlet IDs are ready to be found in the DOM
});
</script>

<!-- rendering is delegated to the specified href for each locale -->

<body class="lotusui tundra locale_en">
  <div class="lotusFrame">
    <div class="lotusui lotusBanner" role="banner">
      <div class="lotusRightCorner">
        <div class="lotusInner">
          <a href="#lotusMainContent" accesskey="S" class="lotusAccess">
            <img src="/portal_dojo/v1.4.3/dojo/resources/blank.gif" alt="Skip to main content link using access key S." />
          </a>
        </div>
        <div class="lotusLogo">
          <span class="lotusAltText">IBM WebSphere Portal</span>
        </div>
      </div>
    </div>
    <div id="ibmStatusBarContainer"><a rel="dynamic-content" href="dyn-cs:id:status@tl:oid:csa2.theme"></a></div>
  </div>
</body>
```
<a rel="dynamic-content" href="dyn-cs:id:tabNav@tl:oid:csa2.theme"></a>

<div style="clear: both;"></div>

<a rel="dynamic-content" href="dyn-cs:id:search@tl:oid:csa2.theme"></a>

</div><!--end titleBar-->

<a rel="dynamic-content" href="dyn-cs:id:pageToolbar@tl:oid:csa2.theme"></a>

<div class="lotusMain" id="lotusMain">
  <!-- pb-pageMode-edit indicates that an iwidget should be lazy-loaded upon entering edit mode for the page -->
  <div class="iw-iWidget iw-Standalone pb-pageMode-edit" id="customizeShelfContainer">
    <a class="iw-Definition" href="/mccbuilder/widget-catalog/customizeShelf.xml?pragma=cache&max-age=1209600&cache-scope=public&vary=none&user-context=false"></a>
  </div>
  <div style="clear:both;"></div>
  <a id="lotusMainContent" name="lotusMainContent"></a>
  <div id="layoutContainers" class="ibmLayoutContainers ibmLayoutContainersHidden" role="main">
    <a rel="dynamic-content" href="dyn-cs:id:layout@tl:oid:csa2.theme"></a>
  </div><!--end main-->
</div><!--end frame-->

<!-- active site analytics additions -->
<a rel="dynamic-content" href="dyn-cs:id:asa@tl:oid:csa2.theme"></a>

<!-- This is responsible for bootstrapping the configuration for the javascript framework. This is located here instead of the head section to improve client performance. -->
<a rel="dynamic-content" href="dyn-cs:id:config@tl:oid:csa2.theme"></a>

<div id="systemWidgets">
  <div class="iw-iWidget iw-Standalone" id="templateLayout">
    <a class="iw-Definition" href="/mccbuilder/widget-catalog/templateLayout.xml?pragma=cache&max-age=1209600&cache-scope=public&vary=none&user-context=false"></a>
  </div>
  <div class="iw-iWidget mumHiddenWidget iw-Standalone" id="pageActionsMenu">
    <a class="iw-Definition" href="/mccbuilder/widget-catalog/ContentSetMenu.xml?pragma=cache&max-age=1209600&cache-scope=public&vary=none&user-context=false"></a>
    <span class="iw-ItemSet" title="attributes" style="display: none;">
      <a class="iw-Item" href="#contextMenuID">pageActions</a>
      <a class="iw-Item" href="#anchorCSSClass">lotusCommonActionMenuAnchor</a>
      <a class="iw-Item" href="#menuCSSClass">lotusCommonActionMenu</a>
    </span>
  </div>
</div>
98:  <a class="iw-Item" href="#resourceType">com.ibm.mashups.enabler.navigation.NavigationNode</a>
99:  <a class="iw-Item" href="#openEvent">PageActions.open</a>
100:  <a class="iw-Item" href="#closeEvent">PageActions.close</a>
101:  </span>
102:  </div>
103:  <div class="iw-iWidget mumHiddenWidget iw-Standalone" id="userActionsMenu">
104:  <a class="iw-Definition" href="/mccbuilder/widget-catalog/ContentSetMenu.xml?pragma=cache&max-age=1209600&cache-scope=public&vary=none&user-context=false"></a>
105:  <span class="iw-ItemSet" title="attributes" style="display: none;"> 
106:  <a class="iw-Item" href="#contextMenuID">userActions</a>
107:  <a class="iw-Item" href="#anchorCSSClass">lotusCommonActionMenuAnchor</a>
108:  <a class="iw-Item" href="#menuCSSClass">lotusCommonActionMenu</a>
109:  <a class="iw-Item" href="#resourceType">com.ibm.mashups.enabler.user.User</a>
110:  <a class="iw-Item" href="#openEvent">UserActions.open</a>
111:  <a class="iw-Item" href="#closeEvent">UserActions.close</a>
112:  </span>
113:  </div>
114:  <div class="iw-iWidget mumHiddenWidget iw-Standalone pb-pageMode-edit" id="newPage">
115:  <a class="iw-Definition" href="/mccbuilder/widget-catalog/NewPageDialog.xml?pragma=cache&max-age=1209600&cache-scope=public&vary=none&user-context=false"></a>
116:  <span class="iw-ItemSet" style="display:none" title="attributes">
117:  <a class="iw-Item" href="#controller">ibmCfg.controllers.navigation</a>
118:  <a class="iw-Item" href="#allowFriendlyURL">true</a>
119:  <a class="iw-Item" href="#allowPrivate">true</a>
120:  <!-- <a class="iw-Item" href="#showPageTemplate">ibmPortalConfig.isContentTemplatingEnabled</a> -->
121:  </span>
122:  </div>
123:  <!-- The lazyLoad attribute is used for widgets that are explicitly loaded 
124:       by another source at a later point in time 
125:       (in this case the displayHelper). -->
126:  <div class="iw-iWidget mumHiddenWidget iw-Standalone" lazyLoad="true" id="sharePage">
127:    </div>
128:  <div class="iw-iWidget mumHiddenWidget iw-Standalone" lazyLoad="true" id="reorderPage">
129:    </div>
130:  <div class="iw-iWidget mumHiddenWidget iw-Standalone" lazyLoad="true" id="viewMorePage">
131:    </div>
Chapter 12. Creating common navigation across the products

Edit css/master.css
This file is the master.css and anything added here will take precedence over other css.

In order to add and position the logo the following line is edited. The text in bold indicates the logo file name and the text that may need to be changed in order to size and position the logo correctly.
In order to make the navigation theme match with that of Connections the following lines are added to the bottom of the file.

```
.lotusBanner ul li a {margin-top:0px; margin-bottom: 0px;}
.lotusBanner ul.lotusLinks li.lotusSelected a {text-decoration: none;}
.lotusNavMenu tr {background-color: transparent;color: #666666;white-space: nowrap;}
.lotusNavMenu th, .lotusNavMenu td {padding: 0 5px;text-align: left;}
.lotusNavMenu tr a, .lotusNavMenu tr span {display: inline-block;padding: 5px;}
.lotusNavMenu tr img {margin: 0 3px;}
.lotusui img {border: 0 none; vertical-align: middle;}
lconnSprite-iconBlogs16 {background-position: 0 -48px; height: 16px; width: 16px;}
lconnSprite-iconActivities16 {background-position: 0 0;height: 16px;width: 16px;}
lconnSprite-iconBookmarks16, lconnSprite-iconDogear16 {background-position: 0 -96px;height: 16px;width: 16px;}
lconnSprite-iconFiles16 {background-position: 0 -192px;height: 16px;width: 16px;}
lconnSprite-iconForums16, lconnSprite-iconForum16 {background-position: 0 -216px;height: 16px;width: 16px;}
lconnSprite-iconWikis16 {background-position: 0 -336px;height: 16px;width: 16px;}
.lotusNavMenu table {list-style-type: none;}
.lotusNavMenu tr {color: #666666;white-space: nowrap;}
.lotusui .lotusNowrap {white-space: nowrap;}
```

12.1.4 References

Using external CSS site templates:

12.2 Customizing IBM Connections theme

In this section we will customize the Connections theme to keep the business standard look and feel across the Connections products. For this we will create a custom theme, add links to Portal and Quickr, along with the default links used for Connections, and update the image just like we are doing in Portal and Quickr.
12.2.1 Customize the navigation bar

To customize the navigation bar:

1. From any of the Connections applications copy the header.jsp into the customization directory
   a. Copy the file from the installed app/component war/nav/templates
      (E:\IBM\WebSphere\AppServer\profiles\AppSrv01\installedApps\connectionsCell01\Homepage.ear\Homepage.war\nav\templates in our environment) to the
      \common\nav\templates directory
      (E:\IBM\LotusConnections\data\shared\customization\common\nav\templates in our environment)

2. Add link to WebSphere Portal welcome page
   a. Open header.jsp in the customization directory in a text editor
   b. Search for the following lines in the file

```html
1:   --%><ul class="lotusInlinelist lotusLinks" role="navigation"
2:     aria-label="<fmt:message key="label.header.navigation.label" />">%--
3:       The Homepage link.
4:     --%>
5:   </li>
```
   c. Enter the following code (marked in bold) between these two lines to add a link to the Portal server: **Note**: We added them here so the link back to the Portal server would be placed before the link to the Connections homepage.

```html
1:   --%><ul class="lotusInlinelist lotusLinks" role="navigation"
2:     aria-label="<fmt:message key="label.header.navigation.label" />">%--
3:     --%>
4:   <li>
6:     </li>
7:   --%>
8:   The Homepage link.
9: 
```
   d. Save and close the file.

3. Add the link to the Lotus Quickr server.
   a. Open header.jsp in the customization directory in a text editor
   b. Search for the following lines in the file:
c. Enter the following code (marked in bold) between these two lines to add a link to the Quickr Domino server: **Note:** We added them here so the link to Quickr Domino would be placed between the Communities and Apps links.

```
1:  --%>
2:  </c:if>
3:  --%
4:        <li>
6: </li>
```

4. Restart Connections and test the URL.

After you restart Connections open a browser to the connections URL:

You should see the new links to WebSphere Portal and Quickr Domino:

![](image)

**12.2.2 Change logo to company image**

We wanted to use a common logo across the products. In our case we used an image that says IBM Redbooks.
1. Copy your company logo image file (for example, CoLogo.gif) into the images subdirectory of the appropriate customization directory. (E:\IBM\LotusConnections\data\shared\customization\common\nav\common\styles\images in our test environment)

2. Rename the logo image to logo.png.

3. Restart the Connections server and delete all temporary internet files on the client browser, the image should now show the new company logo:

12.3 Customization of Lotus Quickr Domino theme

In this section we will customize the Quickr Domino theme to keep the business standard look and feel across the products.

12.3.1 Overview

To customize the Quickr Domino theme to keep the business standard look and feel across the products, we will add links to Portal along with the default links used for Connections, and update the logo just like we are doing in Portal and Connections. With Quickr we have to do this in two locations, first in the Landing Page and then in the Places themselves.

Here we show the Quickr Domino landing page before customization with its default banner, blue in colour, the Lotus Quickr logo and no links.

Here we have a Quickr Domino place banner before customization which is like the landing page banner.

12.3.2 Customize the navigation bar in the Quickr Domino landing page

In this step we will customize the navigation bar for the Quickr Domino landing page. This involves creating and populating three files (setup_ext.js, addLinks.js and stylesheet.css).

To customize the navigation bar for the Quickr Domino landing page:

1. Create the following empty files in these locations.
2. In `setup_ext.js` we load our new stylesheet to change the look and feel of the banner. Add the following code:

```javascript
/* Load the Stylesheet */
var linkStyle = dojo.create("link");
dojo.attr(linkStyle, {
  rel: "stylesheet",
  href: "/qphtml/skins/setup/stylesheet.css"
});
dojo.query('head')[0].appendChild(linkStyle);
```

3. In `stylesheet.css` we then modify the look of the banner. Add the following code:

```css
/* Copyright IBM Corp. 2010 All Rights Reserved. */
/* nothing */
/*** B A N N E R ***/
/*this is the very top of the UI, that contains the logo, application links, user name, login/logout*/
#lotusBanner {padding:0 5px 0 15px; background:#34363a; border-bottom:1px solid #1a1a1a; height:33px; background:-moz-linear-gradient(top, #222 0%, #34363a 100%); background:-webkit-gradient(linear, left top, left bottom, color-stop(0%, #222), color-stop(100%, #34363a));}
#lotusBanner {margin-bottom:-3px;}
#lotusBanner, .lotusui_ie8 .lotusBanner {margin-bottom:-1px;}
#lotusBanner #lotusLogo {outline:none;}
#lotusBanner #lotusLogo img {margin-top:13px; width:40px;height:15px;background-position:-218px -9px;}
#lotusBanner #lotusLogo img {margin-left:0;}
#lotusBanner #lotusLogo img {margin-top:10px;}
#lotusBanner #lotusLogo #lotusAltText {padding:0; margin:11px 20px 10px 0; color:#ccc; text-shadow:0 1px 2px #000000;}
#lotusBanner a.lotusLogo:hover, #lotusBanner a.lotusLogo:focus, #lotusBanner a.lotusLogo:active {color:#ccc; cursor:pointer/*for IE6/7*/;}
#lotusBanner ul.lotusInlinelist li {margin:0;}
#lotusBanner ul.lotusInlinelist li {display:block; float:left;}
#lotusBanner ul.lotusInlinelist li a {color:#EEEEEE; font-weight:bold; display:block;}
```

210
Next we need to add the new links and style them. To make Quickr Domino look more like portal and Connections, we also add in a drop-down Apps menu for accessing the Connections applications. We need to load this file in the setup_ext.js similar to what we did for the stylesheet, so append the following code to the bottom of the setup_ext.js file.

```javascript
/* Load the New links */
QuickrGeneralUtil.loadSupplementalScript("/qphtml/skins/setup/scripts/addLinks.js");
```
Then inside the addLinks.js file we add the code below. Note: You will need to add your own urls for your links.

```javascript
/* Copyright IBM Corp.  2011  All Rights Reserved. */
/*                                                                   */
/*********************************************************************/

Setup_addLinks = {
  init: function() {
    dojo.addOnLoad(
      function() {
        if(document.createStyleSheet) {
          document.createStyleSheet().addRule('.cssClass', 'color: #EEEEEE;
          border-width: 1px; border-color: black; vertical-align: top;');
          document.createStyleSheet().addRule('.cssClass td', 'vertical-align:
          top !important;');
          document.createStyleSheet().addRule('.cssClass a', 'color: #05386B
          !important; text-shadow: 0 1px 2px #FFFFFF !important; ');
          document.createStyleSheet().addRule('#PlacesOrFiles
          .lotusFirst.lotusSelected', 'display: none;');
          document.createStyleSheet().addRule('#lotusBanner ul.lotusLinks li
          a:focus', 'background: none !important;');
          document.createStyleSheet().addRule('#lotusBanner ul.lotusLinks li
          a:hover', 'background: none !important;');
          document.createStyleSheet().addRule('#pageTitlePlaces', 'display:
          none;');
          document.createStyleSheet().addRule('#popup a', 'padding: 2px
          !important;');
          document.createStyleSheet().addRule('#typePlaces > a', 'display: none
          !important;');
          document.createStyleSheet().addRule('.lotusInner > ul li.lotusFirst',
          'margin-top: 3px !important;');
        } else {
          var style = document.createElement('style');
          style.type = 'text/css';
          style.innerHTML = '
          .cssClass { color: #EEEEEE; border-width: 1px;
          border-color: black; } .cssClass a { color: #05386B !important;
          text-shadow: 0 1px 2px #FFFFFF !important; } #PlacesOrFiles
          .lotusFirst.lotusSelected { display: none; } #lotusBanner ul.lotusLinks li
          a:focus { background: none !important; } #lotusBanner ul.lotusLinks li
          a:hover { background: none !important; }
          #pageTitlePlaces { display:none; } #popup a { padding: 2px !important; }
          #typePlaces > a { display: none !important; } .lotusInner > ul li.lotusFirst
          { margin-top: 3px !important; }';
          document.getElementsByTagName('head')[0].appendChild(style);
        }
      }
    
    Setup_addLinks.addHeaderLinks();
  }
```

Chapter 12. Creating common navigation across the products

```javascript
36: };
37: },
38: },
39: mouseOver: function() {
40: document.getElementById("popup").style.visibility = "visible";
41: Setup_addLinks.setLinks();
42: },
43: mouseOut: function() {
44: document.getElementById("popup").style.visibility = "hidden";
45: },
46: setLinks: function() {
47: document.getElementById("bookmarklink1").href = "http://connections.demos.ibm.com/dogear";
50: document.getElementById("activitylink1").href = "http://connections.demos.ibm.com/activities";
```
```javascript
73: },
74: addHeaderLinks: function() {
75: var placesLink = document.getElementById('typePlaces');
76: if (placesLink) {
77: var childElement = placesLink.firstChild;
78: function buildLink(linkName, linkPath, linkClass) { // Build each link
79: var newLink = document.createElement('li');
80: newLink.id = 'lotusBanner' + linkName;
81: newLink.className = linkClass;
82: newLink.innerHTML = '<a href="http://' + linkPath + '"><FONT COLOR="#EEEEEE">" +linkName +"</a>";
83: return newLink;
84: }
85: var linkParent = childElement.parentNode;
86: var divblock = document.createElement("span");
87: divblock.innerHTML = '<span id="popOver" onmouseover="Setup_addLinks.mouseOver()" onmouseout="Setup_addLinks.mouseOut()" style= "position: absolute; color: #EEEEEE; margin-top: 3px; font-weight: bold;"> Apps </span>';  
88: var popDiv = document.createElement("div");
89: popDiv.innerHTML = '<div id="popup" style = "border: 1px solid black; z-index: 200000; visibility: hidden; margin-top: 17px; position: absolute; top: 5px;left:595px" onmouseover="Setup_addLinks.mouseOver()" onmouseout="Setup_addLinks.mouseOut()"> <table class = "cssClass" style = "position:relative; display:block; top:0px; left:0px; background: white; border-color: black; border-width: 3px; "> <tr><td><b><a id = "bloglink1">Blogs</a></b></td><td><a id = "bloglink2">Latest Entries</a></td></tr><tr><td><b><a id = "bookmarklink1">Bookmarks</a></b></td><td><a id = "bookmarklink2">Popular</a></td></tr><tr><td><b><a id = "activitylink1">Activities</a></b></td><td><a id = "activitylink2">To Do List</a></td></tr><tr><td><b><a id = "filelink1">Files</a></b></td><td><a id = "filelink2">Shared With Me</a></td></tr><tr><td><b><a id = "forumlink1">Forums</a></b></td><td><a id = "forumlink2">I'm an Owner</a></td></tr><tr><td><b><a id = "wikilink1">Wikis</a></b></td><td><a id = "wikilink2">I'm an Owner</a></td></tr><tr><td><a id = "wikilink3">Public Wikis</a></td></tr></table></div>';  
90: linkParent.insertBefore(buildLink("Intranet - Welcome", "portal.demos.ibm.com/wps/myportal", "lotusFirst"), childElement);
91: linkParent.insertBefore(buildLink("Home", "connections.demos.ibm.com/homepage", "lotusFirst"), childElement);
92: var popDiv = document.createElement("div");
93: popDiv.innerHTML = '<div id="popup" style = "border: 1px solid black; z-index: 200000; visibility: hidden; margin-top: 17px; position: absolute; top: 5px;left:595px" onmouseover="Setup_addLinks.mouseOver()" onmouseout="Setup_addLinks.mouseOut()"> <table class = "cssClass" style = "position:relative; display:block; top:0px; left:0px; background: white; border-color: black; border-width: 3px; "> <tr><td><b><a id = "bloglink1">Blogs</a></b></td><td><a id = "bloglink2">Latest Entries</a></td></tr><tr><td><b><a id = "bookmarklink1">Bookmarks</a></b></td><td><a id = "bookmarklink2">Popular</a></td></tr><tr><td><b><a id = "activitylink1">Activities</a></b></td><td><a id = "activitylink2">To Do List</a></td></tr><tr><td><b><a id = "filelink1">Files</a></b></td><td><a id = "filelink2">Shared With Me</a></td></tr><tr><td><b><a id = "forumlink1">Forums</a></b></td><td><a id = "forumlink2">I'm an Owner</a></td></tr><tr><td><b><a id = "wikilink1">Wikis</a></b></td><td><a id = "wikilink2">I'm an Owner</a></td></tr><tr><td><a id = "wikilink3">Public Wikis</a></td></tr></table></div>';  
94: divblock.appendChild(popDiv);
95:  
96: divblock.appendChild(popDiv);  
97:  
98: linkParent.insertBefore(buildLink("Intranet - Welcome", "portal.demos.ibm.com/wps/myportal", "lotusFirst"), childElement);
99: linkParent.insertBefore(buildLink("Home", "connections.demos.ibm.com/homepage", "lotusFirst"), childElement);
```
101: linkParent.insertBefore(buildLink("Profiles", "connections.demos.ibm.com/profiles", "lotusFirst"), childElement);
102: linkParent.insertBefore(buildLink("Communities", "connections.demos.ibm.com/communities", "lotusFirst"), childElement);
103: linkParent.insertBefore(buildLink("Project Places", "quickr.demos.ibm.com/lotusquickr", "lotusFirst"), childElement);
104: linkParent.insertBefore(divblock, childElement);
105: }
106: }
107: }
108: Setup_addLinks.init();

5. Test the URL by deleting all temporary internet files and refreshing the browser.
   You should see the new banner styling and the new links to WebSphere, Portal and Lotus Quickr.

12.3.3 Customize the navigation bar in a Quickr Domino place

In this step we will customize the navigation bar for all Quickr Domino places: This involves creating and populating three files (quickr_ext.js, addLinks.js and stylesheet.css). Configured copies of these files are attached to this document.

To customize the navigation bar for all Quickr Domino places:
1. Create the following empty files in these locations.

<table>
<thead>
<tr>
<th>File</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>quickr_ext.js</td>
<td>DominoDataDir\dominohtml\skins\quickr\scripts</td>
</tr>
<tr>
<td>addLinks.js</td>
<td>DominoDataDir\dominohtml\skins\quickr\scripts</td>
</tr>
<tr>
<td>stylesheet.css</td>
<td>DominoDataDir\dominohtml\skins\quickr\scripts</td>
</tr>
</tbody>
</table>

**Note:** The contents of addLinks.js and stylesheet.css here are similar to those created for the landing page customization but they are not identical.

2. In the **quickr_ext.js** we load our new stylesheet to change the look and feel of the Banner..

   1:
   2: /* Load the Stylesheet */
   3:
   4: var linkStyle = dojo.create("link");
   5: dojo.attr(linkStyle, {
   6:   rel:"stylesheet",
   7:   href:"/qphtml/skins/quickr/stylesheet.css"
   8: });
   9:
3. In the Stylesheet we then modify the look of the Banner.

```css
/* Copyright IBM Corp. 2011 All Rights Reserved. */

/*** B A N N E R **/
/*this is the very top of the UI, that contains the logo, application links, user name, login/logout*/

.lotusBanner {padding:4px 5px 0 15px; background:#34363a; border-bottom: 1px solid #1a1a1a; height:30px; background:-moz-linear-gradient(top, #222 0%, #34363a 100%); background:-webkit-gradient(linear, left top, left bottom, color-stop(0%, #222), color-stop(100%, #34363a));}
.lotusBanner {margin-bottom:-3px;}
.lotusBanner, .lotusui_ie8 .lotusBanner {margin-bottom:-1px;}
.lotusBanner ul.lotusInlinelist li {margin:0;}
.lotusBanner ul.lotusInlinelist li {display:block; float:left;}
.lotusBanner ul.lotusInlinelist li a {color:#EEEEEE; font-weight:bold;}
.lotusBanner ul.lotusLinks li .lotusSelected a {text-decoration:none;}
.lotusBanner ul li a {padding:6px 7px 10px;}
.lotusBanner ul.lotusUtility li a {padding:12px 10px 11px;}
.lotusBanner ul.lotusInlinelist li.lotusUser a {padding-right:10px;}
.lotusBanner ul.lotusInlinelist li { border-left-width: 0px;}
.lotusBanner .lotusDropDownSprite {background-position:0 -656px;}
.lotusBanner .lotusUnreadBadge {background:#ccc; color:#000; text-shadow:0 1px 2px #ccc;}
.lotusBanner ul.lotusUtility a.lotusShare {margin-top:10px;}
.lotusBanner ul.lotusUtility a.lotusShare:hover, .lotusBanner ul.lotusUtility a.lotusShare:focus {background:#fff;}
.a.lotusShare:hover .lotusShareConnector, .a.lotusShare:focus .lotusShareConnector {background-position:-964px -13px; width:7px; height:7px;}
.lotusBanner {overflow: visible;}
.lotusBanner ul.lotusInlinelist a {text-shadow:0 1px 2px #000;}
.lotusBanner ul.lotusLinks li.lotusHover a, .lotusBanner ul.lotusLinks li a:hover, .lotusBanner ul.lotusLinks li a:focus {color:#fff; text-shadow:0 1px 1px #000; background:#434549; background:-moz-linear-gradient(#525459, #434549); background:-webkit-gradient(linear, left top, left bottom, from(#525459), to(#434549));}
```
4. Next up we need to add the new links and style them as we did for the landing page. This time we append the code to the quickr_ext.js.

```javascript
/* Load the New links */
self.QuickrSupportUtil._loadSupplementalScript("/qphtml/skins/quickr/scripts/addLinks.js");
```

5. Then inside add Links.js we add the code below. Note: You will need to add your own urls for your links.

```javascript
/* Copyright IBM Corp. 2007, 2010 All Rights Reserved. */
/*********************************************************************/
Quickr_addLinks = {
    init: function() {
        dojo.addOnLoad(function() {
            if(document.createStyleSheet) {
                document.createStyleSheet().addRule('.cssClass', 'color: #EEEEEE; border-width: 1px; border-color: black;');
                document.createStyleSheet().addRule('.cssClass a', 'color: #05386B !important; text-shadow: 0 1px 2px #FFFFFF !important; ');
                document.createStyleSheet().addRule('#PlacesOrFiles .lotusFirst.lotusSelected', 'display: none;');
                document.createStyleSheet().addRule('#pagePlaceBar', 'z-index: 50;');
                document.createStyleSheet().addRule('#pageTitleBar', 'z-index: -50;');
            } else {
                var style = document.createElement('style');
                style.type = 'text/css';
                style.innerHTML = '.cssClass { color: #EEEEEE; border-width: 1px; border-color: black; } .cssClass a { color: #05386B !important; text-shadow: 0 1px 2px #FFFFFF !important; } #PlacesOrFiles .lotusFirst.lotusSelected { display: none; }';
                document.getElementsByTagName('head')[0].appendChild(style);
            }
        });
    }
};
```
26:     document.getElementById('quickr_widgets_misc_myplaces_0').style.display = 'none';
27:
28:     Quickr_addLinks.addHeaderLinks();
29:
30:   }
31: },
32:
33:
34:   mouseOver: function() {
35:     document.getElementById("popup").style.visibility = "visible";
36:     Quickr_addLinks.setLinks();
37:   },
38:
39:   mouseOut: function() {
40:     document.getElementById("popup").style.visibility = "hidden";
41:   },
42:
43:   setLinks: function() {
44:     document.getElementById("bookmarklink1").href = "http://connections.demos.ibm.com/dogear";
47:
51:
55:
59:
60:     document.getElementById("forumlink1").href = "http://connections.demos.ibm.com/forums";
63:
64:  document.getElementById("wikilink1").href =  "http://connections.demos.ibm.com/wikis";
67:  
68: },
69:  
70:  addHeaderLinks: function() {
71:    var placesLink = document.getElementById('PlacesOrFiles');
72:    if (placesLink) {
73:      var childElement = placesLink.firstChild;
74:      function buildLink(linkName, linkPath, linkClass) { // Build each link
75:        var newLink = document.createElement('li');
76:        newLink.id = 'lotusBanner' + linkName;
77:        newLink.className = linkClass;
78:        newLink.innerHTML = '<a href="http://'+ linkPath + '"><FONT COLOR="#EEEEEE">" +linkName +"</a>";
79:        return newLink;
80:      }
81:      
82:      var linkParent = childElement.parentNode;
83:      var divblock = document.createElement("span");
84:      divblock.innerHTML = '<span id="popOver" onmouseover="Quickr_addLinks.mouseOver()" onmouseout="Quickr_addLinks.mouseOut()" style="position: absolute; color: #EEEEEE; margin-top: 2px; font-weight: bold;"> Apps </span>;'
85:      
86:      //divblock.innerHTML = '<span id="popOver" onmouseover="Quickr_addLinks.mouseOver()" onmouseout="setTimeout("Quickr_addLinks.mouseOut()", 10000)" style="position: absolute; color: #CCCCCC; margin-top: 13px; font-weight: bold;"> Apps </span>;'
87:      
88:      var popDiv = document.createElement("div");
89:      popDiv.innerHTML = '<div id="popup" style = "border: 1px solid black; z-index: 200000; visibility: hidden; margin-top: 16px; position: absolute; top: 5px;left:622px" onmouseover="Quickr_addLinks.mouseOver()" onmouseout="Quickr_addLinks.mouseOut()" style = "position:relative; display:block; top:0px; left:0px; background: white; border-color: black; border-width: 3px; ">
90:        <tr><td><a id = "bloglink1">Blogs</a></td><td><a id = "bloglink2">Latest Entries</a></td>
91:        <td><a id = "bookmarklink1">Bookmarks</a></td><td><a id = "bookmarklink2">Popular</a></td>
92:        <td><a id = "activitylink1">Activities</a></td><td><a id = "activitylink2">To Do List</a></td>
93:        <td><a id = "filelink1">Files</a></td><td><a id = "filelink2">Shared With Me</a></td>
94:        <td><a id = "forumlink1">Forums</a></td><td><a id = "forumlink2">I'm an Owner</a></td>
95:      </tr><tr style = "width:150px;"><td><a id = "activitylink3">High Priority Activities</a></td><td><a id = "filelink3">Pinned Folders</a></td>
96:      </tr><tr style = "width:150px;"><td><a id = "forumlink3">Public Folders</a></td>
97:      

6. Test the URL, delete all temp internet files and refresh the browser.

You should see the new banner styling and the new links to WebSphere, Portal and Lotus Quickr.

![New Menus and Styling](image)

### 12.3.4 Change logo to company image in the Quickr Domino Landing page

We wanted to use a common logo across the products. In our case we used an image that says IBM Redbooks.

1. Copy your company logo image file (for example, `CoLogo.gif`) into the images subdirectory of the appropriate customization directory.
   (C:\Lotus\Domino\data\domino\html\qph.html\skins\setup in our test environment)

2. Add the code for the new logo customization file to the bottom of `setup_ext.js`.

```javascript
/* Load the new logo */
QuickrGeneralUtil.loadSupplementalScript("/qph.html/skins/setup/scripts/newLogo.js");
```

3. Create the new logo customization file and add the code below into the file, this will replace the current logo with the new logo.
1: var Setup_replaceLogo = {
2:  main: function() {
3:    dojo.addOnLoad(function() {
4:      var logo = dojo.byId("lotusLogo");
5:      if(logo != null) {
6:        logo.src = "/qphtml/skins/setup/CoLogo.gif";
7:      }
8:    });
9:  }
10: }
11: }
12: }}
13: }
14: Setup_replaceLogo.main();

4. To adjust the sizing of the image, resize your file to approximately 93x17px to fit into the current logo size.
   Delete all temp internet files and refresh the browser, the image should now show the new company logo.

12.3.5 Change logo to company image in a Quickr Domino place

Next we want to add the logo to the Quickr places as well, this is added in a different location.
1. Copy your company logo image file (for example, CoLogo.gif) into the images subdirectory of the appropriate customization directory.
   (C:\Lotus\Domino\data\domino\html\qphtml\quickr\setup\images in our test environment)
2. Load the new logo customization file into the Setup_ext.js (as mentioned above).

1: /* Load the new logo */
3: 
4. Create the new logo customization file and add the code below into the file, this will replace the current logo with the new logo.

1: var Quickr_replaceLogo = {
2:  main: function() {
3:    dojo.addOnLoad(function() {
4:      var logo = dojo.byId("lotusLogo");
5:      if(logo != null) {
6:        logo.src = "/qphtml/skins/setup/CoLogo.gif";
7:      }
8:    });
9:  }
10: }
11: Quickr_replaceLogo.main();
12.4 Navigating from iNotes to other products

This section describes navigation from iNotes to other products.

12.4.1 Overview

There is no banner customization facility which allows iNotes to provide navigation links to other services in the way that this can be done for Portal, Connections and Quickr. However, iNotes widgets can provide a means of connecting to external resources from within an active iNotes window.

12.4.2 Configuring an iNotes widget to open another web site

While mail rules can be created to enable iNotes widgets, one notes.ini parameter may be used to quickly enable or disable iNotes widgets regardless of policies. Place `iNotes_WA_Widgets=1` in the `notes.ini` and restart HTTP. This will enable the widgets folder and allow you to create your own iNotes widget with the sample XML.

The sample XML code below can be modified to point to other resources which you wish to open as separate tabs in iNotes.

1. Right click on Widgets → Create.
2. Increment the `palleteItem id` as iNotes uses this ID to keep track of the widgets.
3. Enter the full URL, the web address, of the site you want.
4. Enter the `title` which the widget will display.
5. Point the `imageUrl` to that site’s favorite icon, favicon.ico, if it has one, and that icon will display for the widget.
6. Save the xml.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<webcontextConfiguration version="1.1">
```
Create other widgets you want to access from your iNotes mail Widgets folder. The Domino administrator may, also, create a widgets catalog.