OpenESP

OpenESP
Admin Guide

Innholdsliste

Introduction

What is OpenESP

OpenESP is an Open Enterprise Search Platform, built on the foundation of Apache™ Solr™, Apache Tomcat™, Apache ManifoldCF™ and and other state of the art open source products. The goal is to help users getting quickly up to speed without the need to understand each single piece of technology in-depth. OpenESP has many best-practices and configurations built right in from the start.

To get involved in the project, head over to https://github.com/openesp/openesp

Document version

This document relates to OpenESP version 0.4, released 2013-XX-XX. Please visit the project website to look for older or newer versions.

OpenESP components

The product consists of the following parts:

- Apache Tomcat servlet container
- Apache Solr enterprise search server
- Apache ManifoldCF connector and security framework
- Apache ZooKeeper (standalone)
- Vifun request parameter tuning tool
- SolrMeter benchmarking tool
- Prepared for Sematext SPM and Search Analytics SaaS
- An OpenESP administration panel for a full platform overview
- Windows installer (IzPack based)

Tomcat

Apache Tomcat (tomcat.apache.org) is an open source web server and servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet specification, required for running Apache Solr.

Solr

Apache Solr (lucene.apache.org/solr) is an open source enterprise search engine. It is the most deployed search engine in the world. Its major features include powerful full-text search, hit highlighting, faceted search, near-real-time search, database integration, and rich document (e.g., Word, PDF) handling. With SolrCloud it easily scales to hundreds of servers, providing full fault tolerance for both search and indexing, with distributed search/indexing, leader election and recovery on failures etc.

ManifoldCF

Apache ManifoldCF (manifoldcf.apache.org) is aproduct which provides an open source framework for connecting source content repositories like Web sites, File systems, Microsoft Sharepoint or EMC Documentum, to target repositories or indexes, such as Apache Solr. ManifoldCF also defines a security model for target repositories that permits them to enforce source-repository security policies. The framework is pluggable, so support for new repositories can be added in a matter of hours or days.

ZooKeeper

Apache ZooKeeper (zookeeper.apache.org) is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services. It is a key requirement for reliably running a SolrCloud cluster. Although Solr can be started with an in-process ZooKeeper, it is recommended to use a standalone cluster of 1, 3 or 5 ZooKeeper nodes.

Other tools

There are many useful 3rd party tools related to Solr. OpenESP bundles a few of these:

Vifun (github.com/jmlucjav/vifun) was developed in order to help tweak the values of boosting functions etc in Solr, typically when using edismax handler.

SolrMeter (code.google.com/p/solrmeter/) is a "generic tool to interact specifically with Solr", firing queries and adding documents to make sure that your Solr implementation will support the real use. With SolrMeter you can simulate your work load over solr index and retrieve statistics graphically.

Sematext SPM (sematext.com/spm/) is a performance monitoring solution available in the Cloud (SaaS) as well as on Premises. Track key operational metrics with a graphical dashboard. We've made it super easy to enable SPM with OpenESP.

Sematext Search Analytics (sematext.com/search-analytics/) is a cloud-based Search Analytics service. It lets you view up-to-the-minute Search Analytics graphs, charts, and tabular data, spot trends, slice and dice data by time, by query origin or type, and other dimensions. Get started in record time with the pre-integrated search GUI.

OpenESP

Versions and release notes

This table lists the versions used and a link to release notes for each component. Please read each of these documents to know about important changes from previous version as well as known bugs and limitations.

Component	Version	Release notes/errata
OpenESP	0.4	https://github.com/openesp/openesp/wiki/Release-notes
Tomcat	7.0.42	http://tomcat.apache.org/tomcat-7.0-doc/changelog.html
Solr	4.4	http://lucene.apache.org/solr/4_4_0/
ManifoldCF	1.3	http://manifoldcf.apache.org/en_US/release-documentation.html
ZooKeeper	3.4.5	http://zookeeper.apache.org/releases.html#releasenotes
SolrMeter	0.3	https://code.google.com/p/solrmeter/wiki/about

System requirements

Hardware requirements

OpenESP runs well both in dedicated and virtualized environments. It can run on your laptop with 100Mb, but here we list adviced mimimum spec for a small-scale production system. Observe that a search server may require extreme performance, so if you rely on shared resources you may not be able to predict the actual performance. In such cases be sure to run on dedicated hardware with dedicated I/O bandwidth.

CPU: Modern, multi-core CPU (64-bit highly recommended, but it will install on 32-bit as well)

RAM: Most search systems will benefit greatly from having 8 Gb or more physical memory, as fitting the whole (or most) of the index in RAM speeds up your search server many-fold.

DISK: For systems with an index size much larger than physical memory, disk performance may be critical, and require high performance disk system, such as multiple 15kRPM+ disks in RAID setup. Alternatively consider SSD. For many systems, the majority of the index fits in RAM and disk thus becomes less important.

Warning: Do not put the index on NFS mounted shared disk. The performance is terrible and it will not work without extensive tweaking.

Final hardware requirements depends on the intended data and query volume. Contact a search specialist for tuning systems with multi-million documents.

Operating System requirements

OpenESP is based on Java technology, and thus in theory runs on any platform, including Linux, Windows and OSX.

Browser requirements

The Administration GUI of Solr uses new standards-based AJAX technology and thus requires a standards-compliant browser. Current versions of any browser should do. Note that if you use Internet Explorer, then at least version 9 is required.

Java requirements

OpenESP runs best with Oracle's official 64-bit JVM. We require Java SE 6 Update 30 or later. If you use Java SE 7, then make sure to use Update 1 or later, as there is a serious bug in the first SE 7 release.

If you have multiple Java environments on the server, make sure to set your JAVA_HOME system environment variable to point to the one you want to use for OpenESP. If you use the installer, it will attempt to select the most appropriate one automatically.

License

OpenESP is licensed under the Apache Software License v2, see http://www.apache.org/licenses/LICENSE-2.0.html. See LICENSE.txt and NOTICE.txt inside the distribution package for details and information about licenses for included sub projects.

Support & training

Professional support subscriptions as well as classroom or on-site training is available from partners in the Open Enterprise Search Network (OpenESN). This is a network of independent search consulting companies using OpenESP as their preferred platform and cooperating on support and training. See www.openesn.net for more information.

Installation

OpenESP in packaged as one release, which can be installed on various platforms either through installer or as a manual install.

Prerequisites

Make sure your server has a supported 64-bit Operating system.

Make sure you have installed a supported version of Java (JRE or JDK) from www.java.com

Download the latest version of OpenESP.

Read the release notes to see any known bugs and limitations.

General installation notes

Memory allocation

The default setting for JVM memory is 2Gb if you do not change it after install. The general rule of thumb is to allocate to the Java VM just enough memory for the application to run, and not more. The reason is that a search engine will benefit hugely from the Operating System's own built-in caching of the disk-based index. If you allocate too much memory to the JVM, the search may actually be slower. You could start with the default 2Gb setting, and increase it if you experience OutOfMemory exceptions.
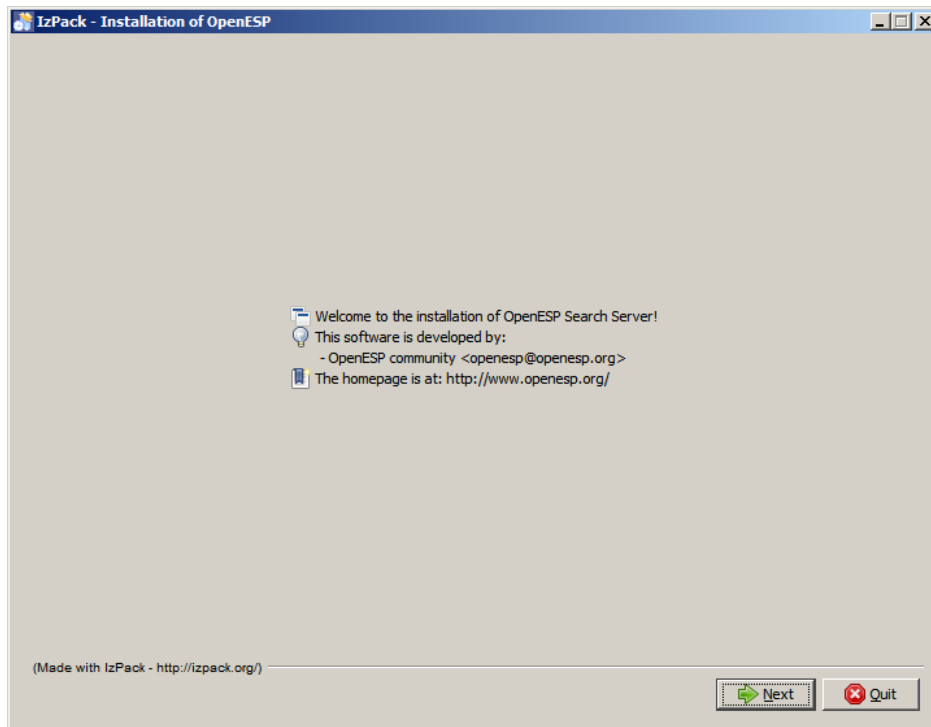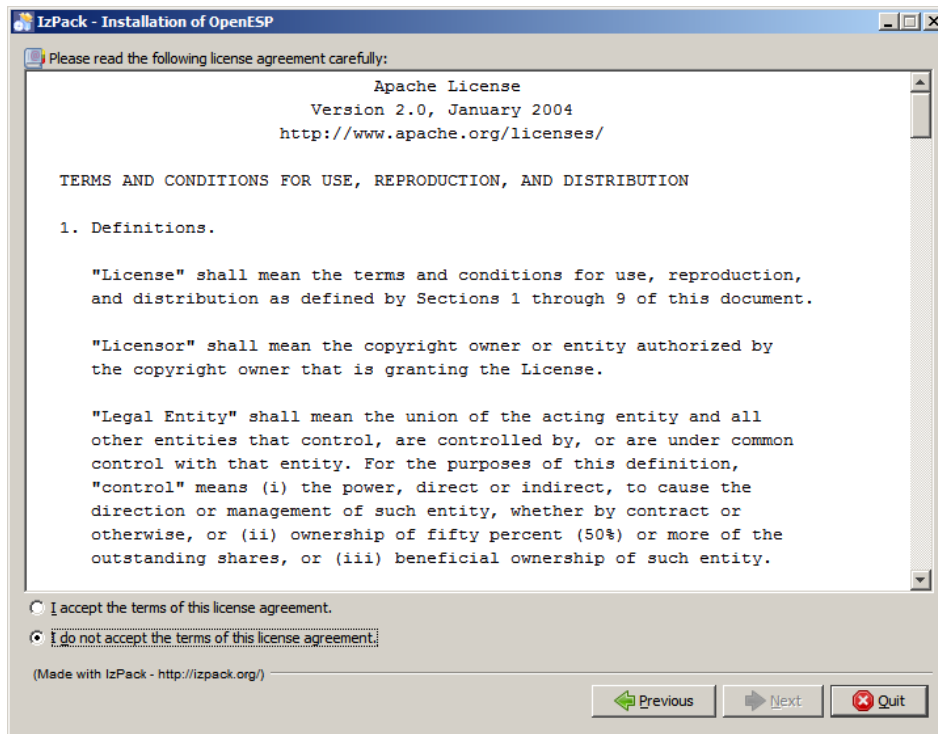
Installing using the installer

The installer is for Windows and Linux types operating systems. The installer will install OpenESP including custom configuration and overlay (if present) and a Windows service or Unix launch-script for automatic start.

Double click `OpenESP-Setup-x.y.exe` or run `java -jar openesp-install-x.y.jar` from command line, and walk through each page of the installer:
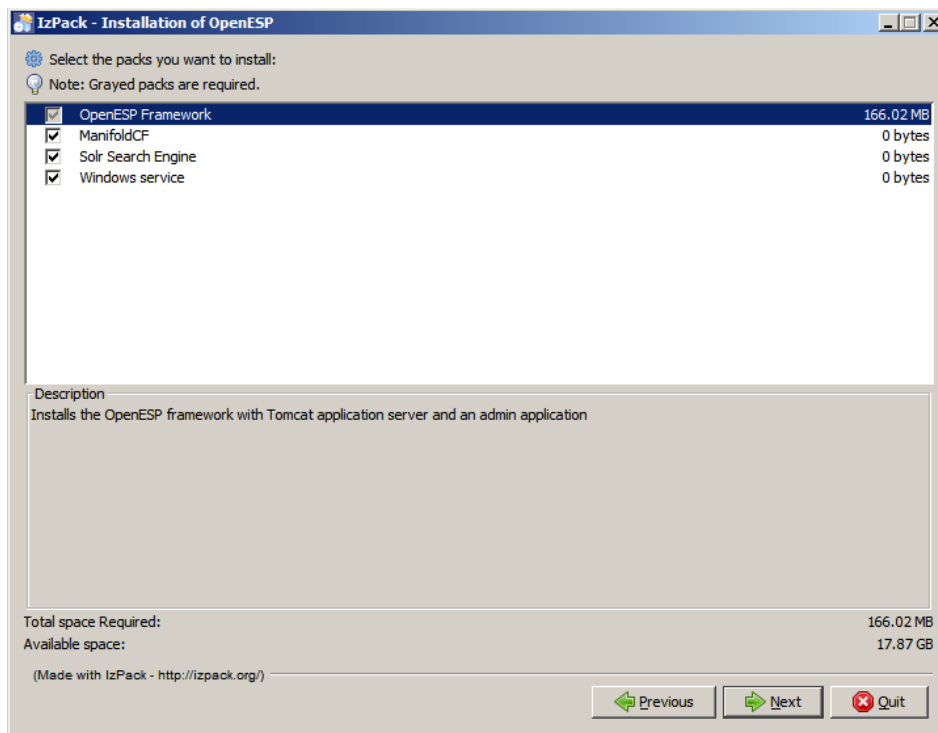
If Java is not installed, the Windows installer will exit and redirect to Oracle's web page for downloading Java. Please accept the license agreement and download the 64-bit version.

NOTE: On a new Windows server, Internet Explorer will complain about non-trusted sites, and not allow any downloads form the internet. The preferred option is to download and install another browser such as Firefox or Chrome. If you need to stick with Internet Explorer, you need to click the "Add" button when promoted, and add a few sites to the whitelist. You also need to go to "Tools->Internet Options->Security->Internet->Custom Level->Downloads->File Download" and select "Enable", to be allowed to download the Java installer.
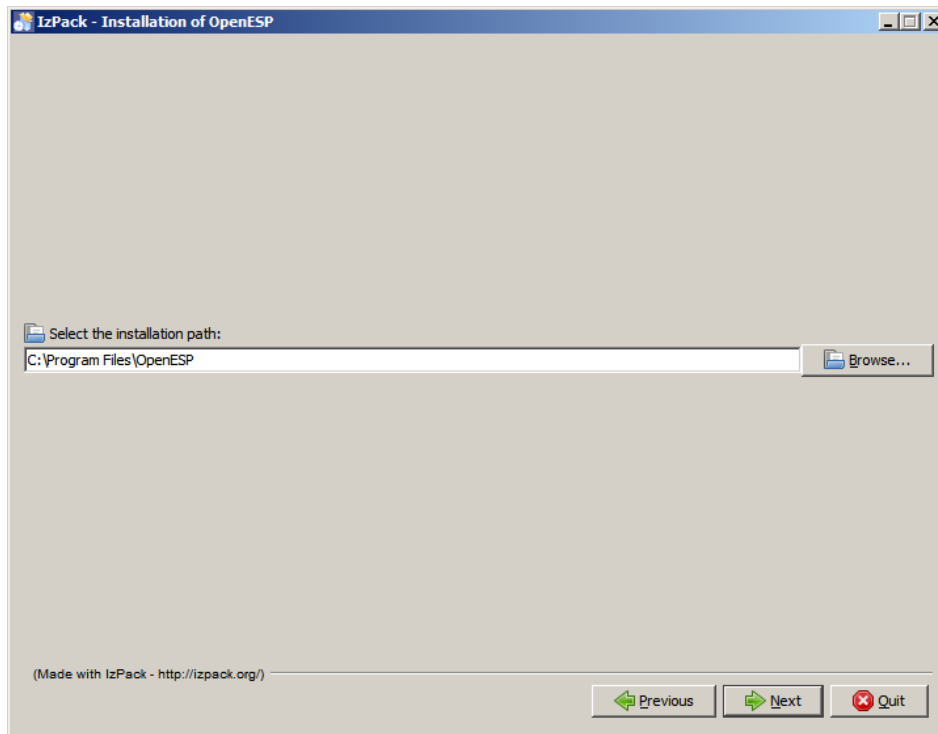
IzPack - Installation of OpenESP

Welcome to the installation of OpenESP Search Server!
This software is developed by:
    - OpenESP community <openesp@openesp.org>
The homepage is at: http://www.openesp.org/

(Made with IzPack - http://izpack.org/)

Next   Quit

IzPack - Installation of OpenESP

Please read the following information:

```
OpenESP (Open Enterprise Search Platform) is a powerful enterprise search server based

The product includes:
* Apache Solr 4
* Apache Tomcat 7
* ManifoldCF 1
* OpenESP administration page

NOTE: We recommend 64 bit OS and Java

The product is open source, released under the Apache Software License.
Read more at http://www.openesp.org/
```

(Made with IzPack - http://izpack.org/)

Previous   Next   Quit

Choose what packages to install.

Choose your installation destination

OpenESP

If your installer comes bundled with a custom Solr configuration zip, the installer can unpack that zip file and use it as its configuration. If a file called openesp-overlay.zip is found in the same directory as the installer, the installer will suggest it for you automatically. Default port is 18080 and default memory is 2048Mb.
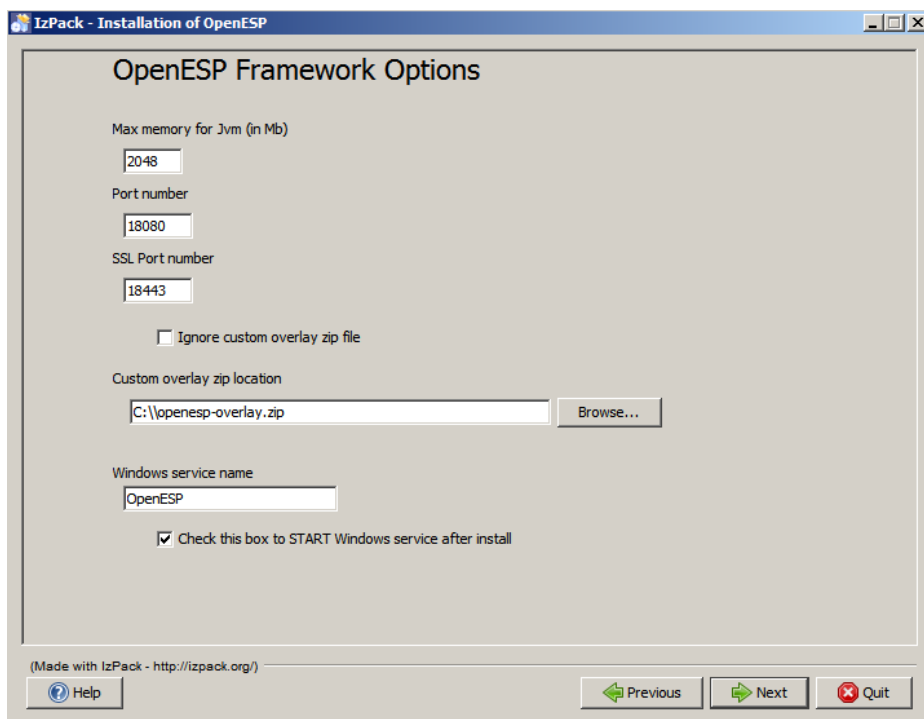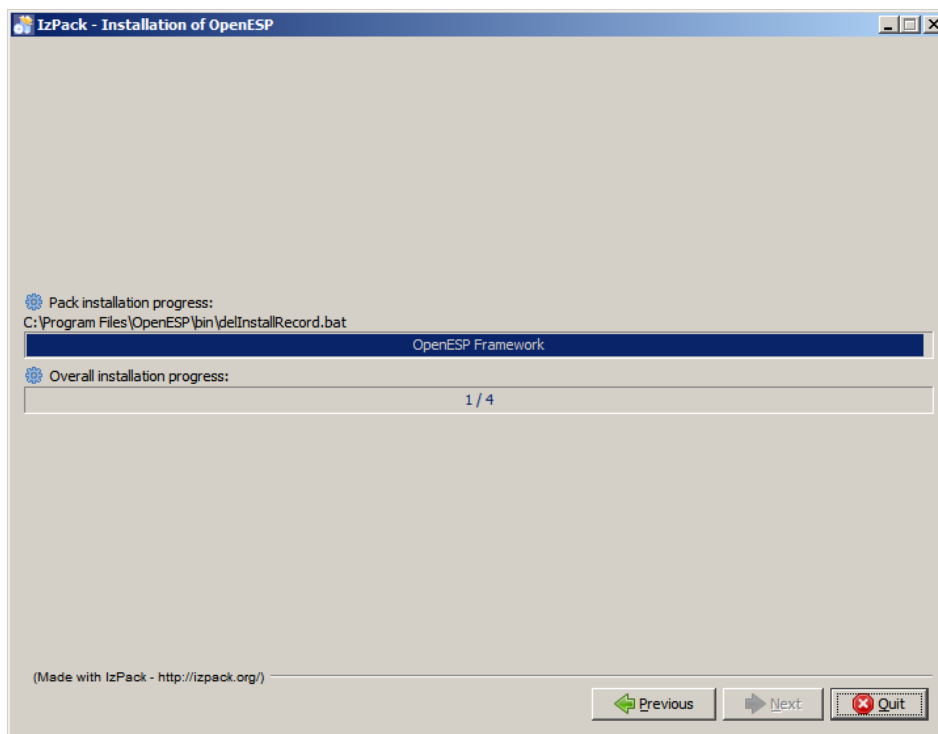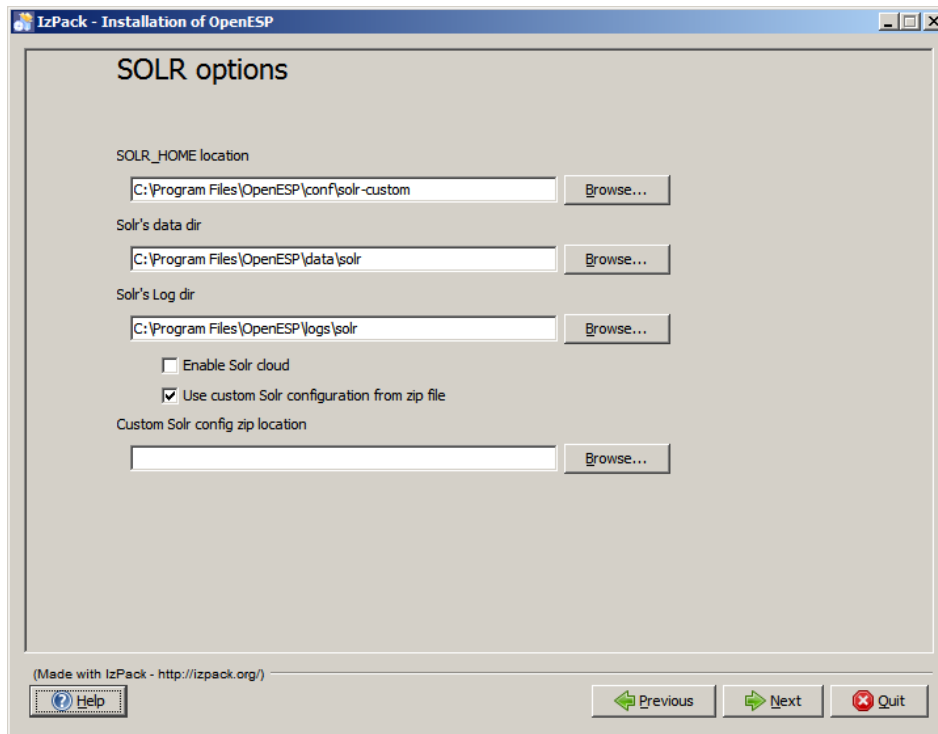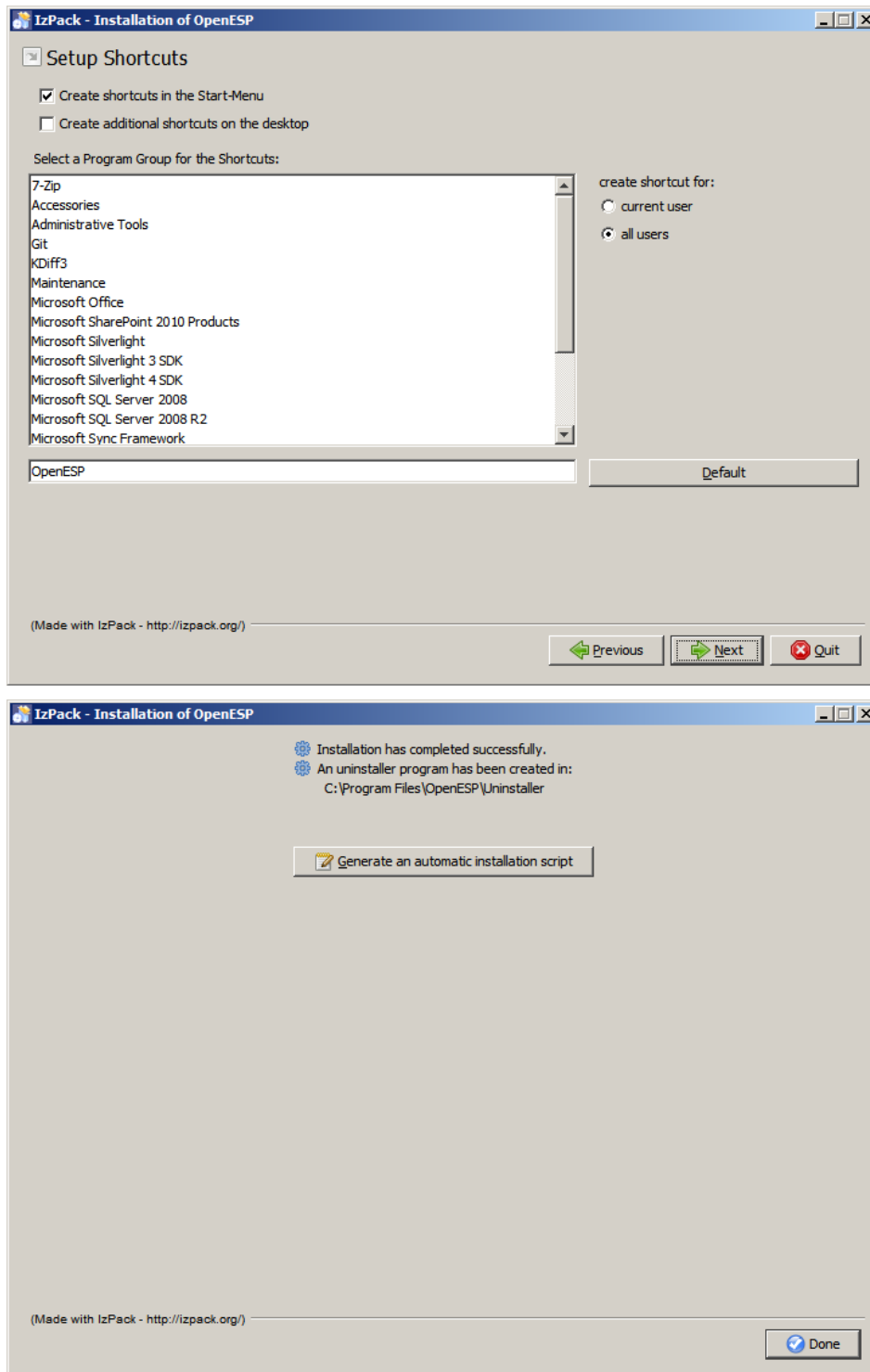
After successful installation, confirm the installation by opening http://localhost:18080/solr/ (replace localhost with your server's host name) in a browser.

Manual installation

If you want to to a manual install or automate the install using tools like Puppet (http://puppetlabs.com/) or Chef (http://www.opscode.com/chef/), follow this procedure. Although we use Unix syntax in this description, Windows users simply replace forward-slash with backslash and *.sh with *.bat

The procedure is simple:

1. Obtain the ZIP deployment package (typically openesp-x.y.zip). After building openesp yourself from source, this resides in build/distributions/

2. Un-zip the deployment package `openesp-x.y.zip` into a location of choice. We call this location `OPENESP_HOME` for convenience

3. Note that ManifoldCF is disabled by default. To enable, open a command prompt, then CD to `$OPENESP_HOME/bin/` and run `./openespctl enable mcf`

4. To start the server for local testing, CD to `$OPENESP_HOME/bin/` then run `./run.sh` or `run.cmd`

5. To start the server in production, install a Windows service or Linux init script. This is currently only supported by installer. (In future there will be tool support through `$OPENESP_HOME/bin/openespctl` for this too.)

6. To use a custom Solr configuration, edit the start script and modify the `SOLR_HOME` environment variable to point to your new solr-home

7. To change memory settings, edit the start script and modify `-Xms` and `-Xmx` settings accordingly.

8. To change port number, open `$OPENESP_HOME/tomcat/conf/server.xml` in an editor and change port number from `18080` to what you need.

9. After successful installation, confirm the installation by opening <ins>http://localhost:18080/solr/</ins> (replace localhost with your server's host name) in a browser.

Configuration

This chapter lists some common configuration options after install of the product. There is a command-line tool to help with many common tasks, and it is located in `bin/openespctl`

General

Ports

OpenESP uses Apache Tomcat as the Java Servlet container. During install the port number is set in `$OPENESP_HOME/tomcat/conf/server.xml.` You may change that to have OpenESP listen on another port.

```
bin/openespctl port <new port number> [<new ssl port number>]
```

‡‡OpenESP

Example: To change to port 8983 after install

```
bin/openespctl port 8983
```

Enable or disable apps

```
bin/openespctl <enable|disable> <app>
```

Example: to enable ManifoldCF (it is disabled by default), run

```
bin/openespctl enable mcf
```

Windows service startup options

When run as a Windows service, startup options such as memory, SOLR_HOME, logging options, which user to run the service as etc are registered in the service daemon program. To view/change these settings, you double click %OPENESP_HOME%\bin\OpenESPw.exe, or:

1. Open a shell/command window with Administrator rights
   [Windows+R] enter "cmd.exe", then hit [SHIFT+CTRL+ENTER] to start as admin

2. CD %OPENESP_HOME%\bin\

3. OpenESPw.exe

4. Do the needed changes

Securing the search server

Sometimes you want to secure the communication with Tomcat/Solr so that unauthorized people cannot access the search index. There are several ways to do that. The easiest is to block OpenESP/Tomcat's port (18080) in the server's firewall, so that traffic is only allowed from certain IP addresses or only from localhost.

Another way is to enable encrypted communication through SSL in Tomcat and only listen on the secure port using server certificates. OpenESP ships with SSL port enabled at port 18443, setup with client and server certificates, already setup in Tomcat. You'll find the certificates in $OPENESP_HOME/tomcat/conf/ssl/ folder.
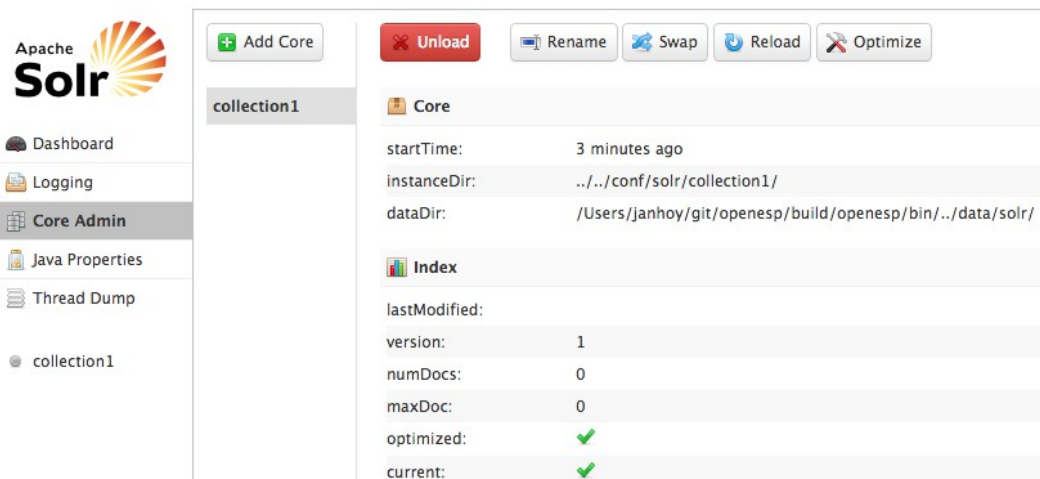
To access the secure port from a browser, you'll need to import the CA certificate ca.pem and the browser certificate browser-client.pfx into your browser.

To re-generate your own certificates, run the script gen-certs.sh

TIP: To block access to insecure port 18080, edit $OPENESP_HOME/tomcat/conf/server.xml

Solr

Solr's configuration resides in $OPENESP_HOME/config/solr. After changing, reload the core to pick up changes. This can be done in Solr's administration panel:

For Solr configuration options, plese refer to the Solr Wiki: http://wiki.apache.org/solr/

ManifoldCF

If you want to push configurations to ManifoldCF the mcf-api-service is the usual endpoint. Since OpenESP uses the "combined" war (http://manifoldcf.apache.org/release/release-1.2/en_US/how-to-build-and-deploy.html#Single-process+deployable+war) the API will be available at http://localhost:18080/mcf/api/.

Monitoring

We make it easy to setup Sematext SPM with OpenESP. On the welcome screen when you start OpenESP you'll find instructions for how to sign up and download SPM.

Once installed, simply run the command:

```
bin/openespctl spm install
```

This will enable monitoring by including the necessary startup options with your OpenESP service/daemon. Within a few minutes you'll be seeing a dashboard like the one below with full overview of key metrics for your search cluster, including alerts when something is wrong.
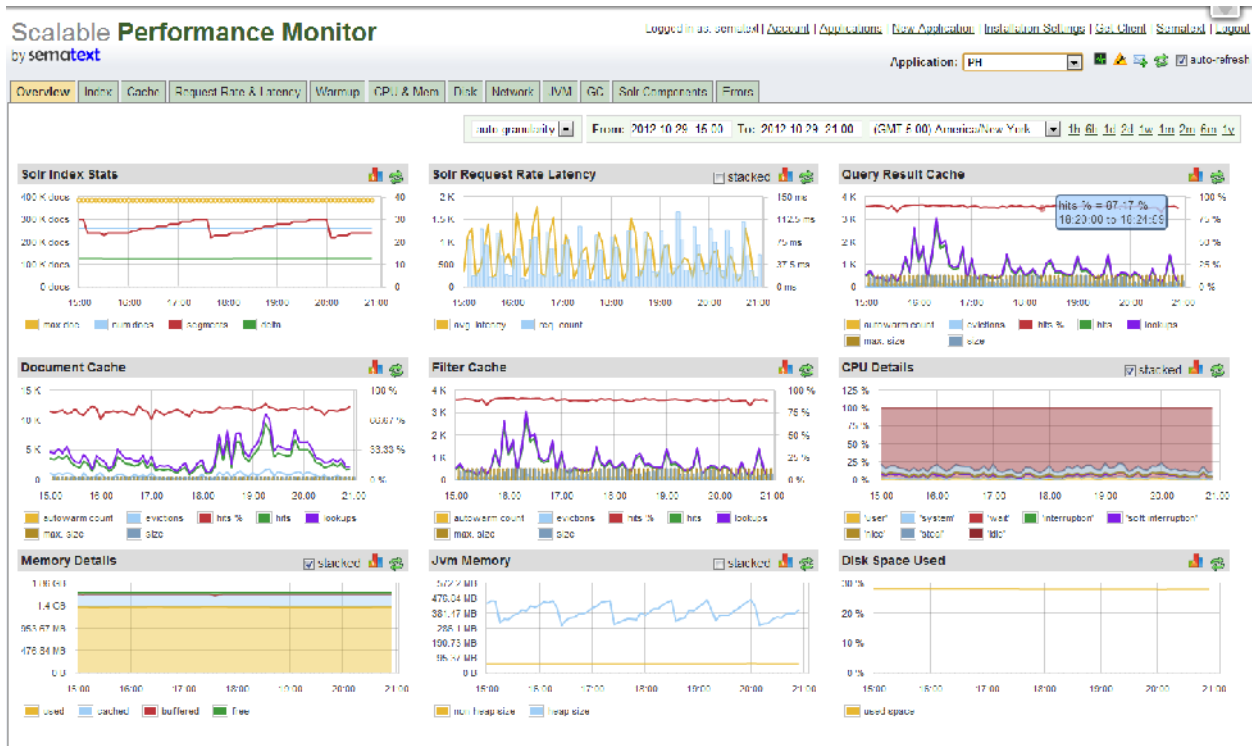
Search Analytics

OpenESP comes pre-bundled with cloud based search analytics integration. It is free to use after you register an account, and gives you deep insight into wow your end users interact with your service, i.e. top searches, identify slow queries and find queries that needs to be tuned because people do not click the top search results. Search Analytics is for your search service what Google Analytics is for your web site.