GO!NotifyLink

Enterprise Server Direct Push Technology

Table of Contents

Mobile Push Technologies

GO!NotifyLink provides users with choices when it comes to mobile device push technology. Users may choose between Scheduled Push or Direct Push, providing the device supports Direct Push.

In **Scheduled Push** mode GO!NotifyLink Device Client software initiates connections to the HTTP/Web server at set intervals. Scheduled Push doesn't use a constant HTTP connection, but rather an opened then closed connection. **Direct Push**, however, has a constant HTTP connection for a period of time.

What is Direct Push Mode?

The direct push technology keeps a device synchronized with data on a GO!NotifyLink server offering the end user faster message delivery.

The direct push technology is implemented with software components residing on both the GO!NotifyLink device client and on the GO!NotifyLink server:

- **GO!NotifyLink Device Client.** The device manages the direct push communication with the GO!NotifyLink server. It establishes an HTTP or HTTPS connection with the server for a specified length of time, and then goes into an energy saving sleep mode while waiting for the server to respond. Server response indicates that new items were received or that no new items arrived. Based on this response, the device sends either a synchronization request or another direct push request. The continuous requests by the device each time the server responds, keep the connection from timing out between updates. The request transmissions are referred to as "heartbeats." Each heartbeat serves to alert the server that the device is ready and still available to receive updates.

- **GO!NotifyLink Enterprise Server (GO!NLES).** On the GO!NotifyLink server, the direct push component delivers e-mail and PIM (contact, calendar, and task) updates to a device as soon as the information arrives at the server using IP-based notifications. The persistent HTTP/HTTPS connection provides the means for up-to-the-minute synchronization for the end user.

How Direct Push Works

Direct push traffic looks like small HTTP requests to an Internet Web site that takes a long time to respond. The direct push process, however, is actually a series of **requests from the device** and **responses from the server**. A direct push session continues until it expires according to a set direct push timeout. (See *Recommendations for Direct Push Settings* below.) The following steps outline the direct push process:

STEP 1: **Initiating Device Request**. The GO!NotifyLink **device client** issues an HTTP(S) message or heartbeat request to the GO!NotifyLink server, asking that the server report any changes that occur in the user's mailbox within a specified time limit.

STEP 2: **Server Response**. When GO!NotifyLink **server** receives the heartbeat request, it monitors the mailbox specified until one of the following occurs:

Occurrence	GO!NotifyLink Server Response
A change occurs in the mailbox.	Issues an HTTP 200 OK response indicating that changes have occurred.
No changes occur in the mailbox.	Issues an HTTP 200 OK response indicating that no changes have occurred.
The time limit expires.	Issues an HTTP 200 OK response indicating that no changes have occurred. **NOTE:** The time limit is determined by the shortest time out in the network path. It is recommended that all time outs in the network path match or exceed the direct push time-out. (See *Recommendations for Direct Push Settings*.)

STEP 3: **Device Reaction to Server Response.** The GO!NotifyLink device client reacts to the response from the GO!NotifyLink server in one of the following ways:

Response From the Server	Device Reaction
An HTTP 200 OK response indicating that changes have occurred.	Issues a synchronization request to the mailbox and then a heartbeat request once the synchronization is completed.
An HTTP 200 OK response indicating that NO changes have occurred.	Re-issues the heartbeat request.
No response from the server within the specified time.	Re-issues the heartbeat request.

Steps 2 and 3 are repeated until the direct push interval expires – a time limit settable on the device by the user or through device policies accessed by an IT administrator on the server.

The device reconnects 60 seconds after the interval expires, establishing another direct push session.

Benefits and Performance

Potential Benefits for the GO!NotifyLink Device Client

Faster message delivery is a major benefit of Direct Push. GO!NotifyLink Server uses IP-based notifications to instantly deliver e-mail and PIM updates to devices as soon as they arrive on your server.

A more cost efficient use of the carrier plan is a second possible benefit. Direct Push synchronization can be more efficient because data is only exchanged when there are updates. Depending on usage and terms of the carrier plan, Direct Push may be the better option for many users.

For some users, increased battery consumption may be a tradeoff associated with Direct Push use. This depends on a host of factors, however, and individual results will vary based on the user. Listed below are some considerations that will give users an idea of the factors that affect device performance and battery life when using Direct Push synchronization.

Performance Considerations

These factors affect the amount of battery power consumed and in turn may impact the user's carrier plan:

- Message Rate (quantity and frequency at which messages arrive)
- Message Notification (ON or OFF)
- Direct Push Timeout Setting
- Device Usage Outside GO!NotifyLink (phone, data)
- Backlight Setting (percentage and timeout)
- Battery Condition/Age
- Data Billing Terms of the Carrier Plan (unlimited vs. metered usage)
- Operator Network (value of idle timeout interval)
- Outgoing Firewall Setting (value of idle timeout interval)

Recommendations for Maximizing Device Performance and Battery Life

- Set Direct Push Timeout Setting to no less than 15 minutes (the default). Also, confirm that outgoing firewall and operator network timeout settings are greater than 15 minutes. – *The longer a connection can remain idle, the less often the device will have to wake up and retransmit, thus reducing battery power and bandwidth consumption.*
- Minimize device backlight settings – *consumes less battery power*
- Set Message Notification to OFF – *reduces data quantity*
- Synchronize essential data folders only - *reduces data quantity*
- Users with heavy message delivery rates (over 300 messages in a 10-hour period) will experience greater battery consumption and greater bandwidth consumption with Direct Push. If your carrier plan allows for unlimited data usage, cost will not be an issue, however, battery life can be greatly affected. Heavy users that still want the advantage of Direct Push's fast delivery may do well to switch their device to Scheduled Push when not expecting critical information in order to conserve battery power. Carrying a backup battery may be another option for users with a heavy message rate.

Recommendations for Direct Push Settings

Direct Push Timeout

When direct push sessions are longer, there are fewer HTTP round trips, less data sent and received, and less power consumed by the device.

A direct push interval that is too short defeats the purpose of direct push technology. Not only do shorter intervals reduce overall connection time, but battery life is shortened as well. Consider this: In an hour's time, a device set with a 5-minute timeout will ping the server 10 times and be connected approximately 50 minutes out of that hour. The same device set with a 15-minute timeout will ping the server 4 times and be connected for approximately 57 minutes.

The length of direct push timeout intervals has no bearing on the functions of the GO!NotifyLink Server's direct push component. Messages are rapidly delivered regardless of the direct push timeout interval length.

Firewall Timeouts

Administrators should set idle connection timeouts on outgoing firewalls to a value that equals or exceeds the direct push timeout.

Firewalls with idle connection timeouts less than the direct push timeout will close the direct push session rendering mail undeliverable until the device reconnects. The result is users who are unsynchronized for long periods of time. See Firewall Configurations for Best Direct Push Performance

Carrier Timeout Intervals

A second consideration is that many carriers implement timeout intervals that range from just a few minutes to almost an hour. The carrier's timeout interval may be less than the firewall timeout your administrator has set or the direct push timeout you have set on the device. In this case, your session will, again, be interrupted rendering the mail undeliverable until reconnection. Check the GO!NotifyLink Knowledge Base article for recommendations based on carrier settings.

Recommendation

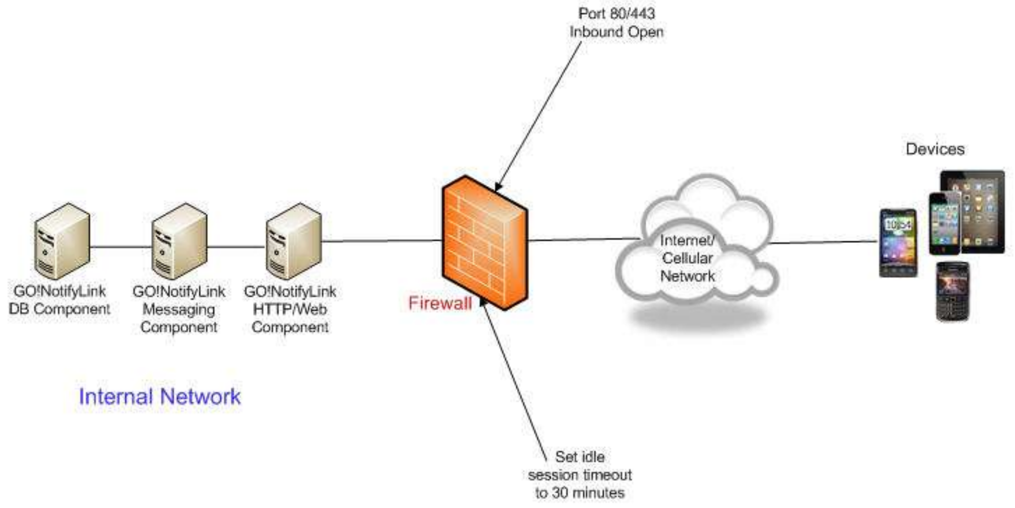
The direct push session length is determined by the shortest timeout in the network path. Therefore, set the device's direct push timeout to the longest possible interval without exceeding the shortest timeout in the network path.

Where to Change the Direct Push Timeout Setting
From a device, select: GO!NotifyLink Preferences > Push Settings
From the Administrative Web, select: User Administration > User IT Policies > Device Policies > Push Settings

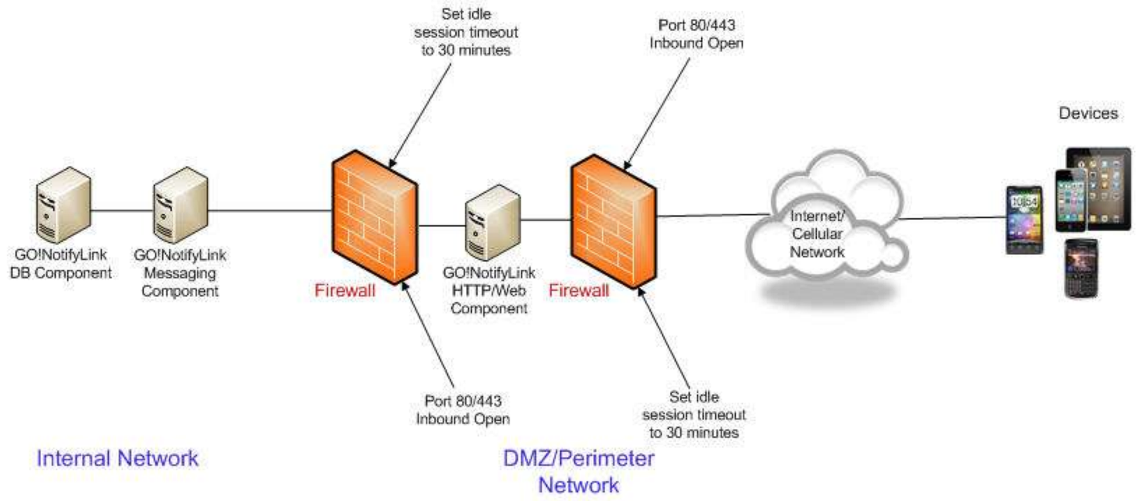
Firewall Configurations for Best Direct Push Performance

The configuration diagrams below show an idle timeout session for the firewalls of 30 minutes. Idle timeout sessions that exceed the direct push timeout will ensure that firewalls do not close the session.

Low idle timeout sessions force a device to reconnect several times, consuming additional battery power. The longer the connection can remain idle, the less often the device will have to wake up and retransmit, and the longer the battery life will be.



Firewall Configuration for Best Direct Push Performance



Firewall Configuration Using a Demilitarized Zone