

# Stored Procedure SQL Server Untuk Mengirim E-mail Dengan Komponen JMail

**Sony Arianto Kurniawan**  
sonyarianto@yahoo.com

## ***Lisensi Dokumen:***

*Copyright © 2003 IlmuKomputer.Com*

*Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.*

Hello pembaca. Anda yang suka bergelut dengan SQL Server tentunya sudah tidak asing lagi dengan operasi-operasi database seperti SELECT, INSERT, UPDATE, DELETE ataupun operasi-operasi manajemen database seperti BACKUP, RESTORE, CREATE dan lain sebagainya. Nah kali ini penulis akan membuat hidup Anda dengan SQL Server lebih hidup lagi dengan membuat suatu stored procedure yang asyik punya. Sebelum kita lanjut maka kita simak dulu pesan-pesan berikut ini.

Tulisan ini bersifat terbuka dan Anda bisa mengirimkan saran dan komentar ke [sonyarianto@yahoo.com](mailto:sonyarianto@yahoo.com). Anda dilarang keras mengutip sebagian atau seluruh tulisan ini tanpa sepengetahuan penulis. Penulis tidak bertanggung jawab atas segala hal yang ditimbulkan akibat tulisan ini.

Sebelum kita mulai maka ada baiknya untuk mengetahui sejenak yang dimaksud dengan stored procedure. Stored procedure adalah sekumpulan perintah-perintah SQL yang tersimpan dalam SQL Server. Procedure ini sudah dalam keadaan ter-compile di server dan siap dipanggil apabila diperlukan. Untuk lebih jelasnya silakan Anda baca buku-buku SQL Server khususnya yang menyangkut masalah stored procedure. OK, kita kembali lagi ke masalah utama. Stored procedure yang akan kita buat kali ini adalah stored procedure untuk mengirimkan e-mail. Di dalam SQL Server sendiri sebenarnya sudah terdapat stored procedure seperti ini yaitu xp\_sendmail, hanya saja lumayan ribet untuk men-setting feature SQL Mail di SQL Server ini. Ribetnya gimana sih? Ya pokoknya ribet deh, sampai-sampai penulis saja masih bingung sampe detik ini.

Kita tinggalkan saja feature SQL Mail ini dan kita akan membuat stored procedure yang lain yaitu dengan memanfaatkan object e-mail w3 JMail dari Dimac. JMail adalah suatu komponen COM berupa file dll yang fungsinya dapat untuk mengirim atau menerima e-mail. Selain itu pada komponen ini terdapat pula fasilitas enkripsi dan mail merge. Dan yang paling penting komponen w3 JMail ini GRATIS, Anda bisa dapatkan di website-nya [www.dimac.net](http://www.dimac.net). Penulis menggunakan w3 JMail versi 4.3. Gimana sih kok SQL Server bisa mengakses komponen JMail untuk mengirim

file? Terimakasih kepada SQL Server yang sudah menyediakan beberapa built-in stored procedure OLE automation. Stored procedure OLE automation di SQL Server adalah:

- sp\_OACreate, untuk membuat instance dari suatu COM object
- sp\_OADestroy, untuk menghapus instance dari suatu COM object
- sp\_OASetProperty, untuk men-set properti dari suatu COM object dengan nilai yang baru
- sp\_OAGetProperty, untuk mendapatkan nilai properti dari suatu COM object
- sp\_OAMethod, untuk memanggil method dari suatu COM object
- sp\_OAStop, wah kalo ini mah gue belon ngerti terjemahannya :) silakan baca help-nya
- sp\_OAGetErrorInfo, untuk mendapatkan informasi error dari OLE automation

Penulis tidak akan membahas stored procedure OLE automation SQL Server pada tulisan ini. Hal itu mungkin akan penulis sajikan pada kesempatan yang lain. Script stored procedure di bawah penulis beri nama sp\_SendMail\_JMail dan memiliki sembilan parameter. Parameter utama yang harus ada yaitu @RecAddr (alamat e-mail tujuan) dan @SMTPServer (alamat SMTP server). Jadi untuk bisa mengirimkan e-mail Anda cukup mengisi dua parameter di atas, tentu saja dengan parameter yang lain akan diisi dengan nilai default masing-masing. Untuk parameter lain penulis kira sudah cukup jelas bagi pembaca (yang sudah mengerti stored procedure) tetapi mungkin masih ada yang bingung antara parameter @Body dan @BodyFromFile. Dua parameter tersebut memiliki fungsi yang sama hanya saja cara pengambilan datanya yang berbeda. Jika Anda ingin menulis langsung body e-mail Anda pada saat menjalankan stored procedure maka Anda bisa menggunakan parameter @Body, contohnya @Body='Ini body e-mail saya...'. Tetapi jika Anda ingin mengisi body e-mail dengan text dari suatu text file di komputer Anda maka Anda bisa menggunakan parameter @BodyFromFile, contohnya @BodyFromFile='c:\bodymail.txt'.

OK, sekarang silakan saja Anda ketik script stored di bawah ini pada *SQL Query Analyzer* atau dapat juga melalui *SQL Server Enterprise Manager*.

```
-- Name: sp_SendMail_JMail
-- Version: 1.00
-- Description: Send mail using w3 JMail free component version 4.3
-- Author: Sony Arianto Kurniawan <sony-ak@aritechdev.com>
-- Home Page: http://www.aritechdev.com/

create procedure sp_SendMail_JMail
@FromAddr varchar(255) = 'sqlserver@aritechdev.com', -- from e-mail
@FromName varchar(255) = 'SQL Server featuring JMail', -- from name
@RecAddr varchar(255), -- recipient address
@RecName varchar(255) = '', -- recipient name
@Attachment varchar(1000) = '', -- attach file path
@Subject varchar(255) = '', -- e-mail subject
@Body varchar(8000) = '', -- e-mail body
@BodyFromFile varchar(1000) = '', -- e-mail body that comes from file
@SMTPServer varchar(255)
as
declare @obj int
declare @res int
declare @ires int
declare @ires2 varchar(255)
exec @res = sp_OACreate 'jmail.Message', @obj out
if @res <> 0
begin
return 1
end
exec @res = sp_OAMethod @obj, 'AddHeader', NULL, 'App-Sender', 'SQL Server
featuring JMail (stored procedure)'
if @res <> 0
```

```
begin
return 1
end
exec @res = sp_OASetProperty @obj, 'From', @FromAddr
if @res <> 0
begin
return 1
end
exec @res = sp_OASetProperty @obj, 'FromName', @FromName
if @res <> 0
begin
return 1
end
if @RecName = ''
begin
exec @res = sp_OAMethod @obj, 'AddRecipient', NULL, @RecAddr
if @res <> 0
begin
return 1
end
end
else
begin
exec @res = sp_OAMethod @obj, 'AddRecipient', NULL, @RecAddr,@RecName
if @res <> 0
begin
return 1
end
end
if @Attachment <> ''
begin
exec @res = sp_OAMethod @obj, 'AddAttachment', @ires2 out, @Attachment
if @res <> 0
begin
return 1
end
end
exec @res = sp_OASetProperty @obj, 'Subject', @Subject
if @res <> 0
begin
return 1
end
if @BodyFromFile <> ''
begin
exec @res = sp_OAMethod @obj, 'AppendBodyFromFile', NULL, @BodyFromFile
if @res <> 0
begin
return 1
end
end
else
begin
exec @res = sp_OASetProperty @obj, 'Body', @Body
if @res <> 0
begin
return 1
end
end
exec @res = sp_OAMethod @obj, 'Send', @ires out, @SMTPServer
```

```
if @res <> 0
begin
return 1
end
if @ires <> 1
begin
return 1
end
return 0
```

Sekarang Anda bisa meng-execute script di atas dan stored procedure akan otomatis tersimpan pada database yang aktif saat Anda mengeksekusi script. Mari kita coba stored procedure yang baru kita buat ini. Penulis akan memberi contoh untuk mengirim e-mail ke alamat [ivani@aritechdev.com](mailto:ivani@aritechdev.com) dengan file attachment 'c:¥data¥mylove.gif' dengan menggunakan SMTP Server milik AriTech Development Indonesia ([mail.aritechdev.com](mailto:mail.aritechdev.com)). Berikut ini adalah contohnya:

```
declare @hasil
exec @hasil = sp_SendMail_JMail
@RecAddr='ivani@aritechdev.com', @Subject='Kepada Esa Ivani',
@SMTPServer='mail.aritechdev.com',@Body='Hello ... ini data file yang
diminta...',@Attachment='c:¥data¥mylove.gif'
```

Pada contoh diatas variabel @hasil akan menampung hasil kembalian dari eksekusi sp\_SendMail\_JMail. Nilai variabel @hasil akan bernilai 1 apabila gagal dan 0 apabila sukses mengirimkan e-mail.

Jika ada kritik atau komentar silakan kirim ke [sonyariato@yahoo.com](mailto:sonyariato@yahoo.com).