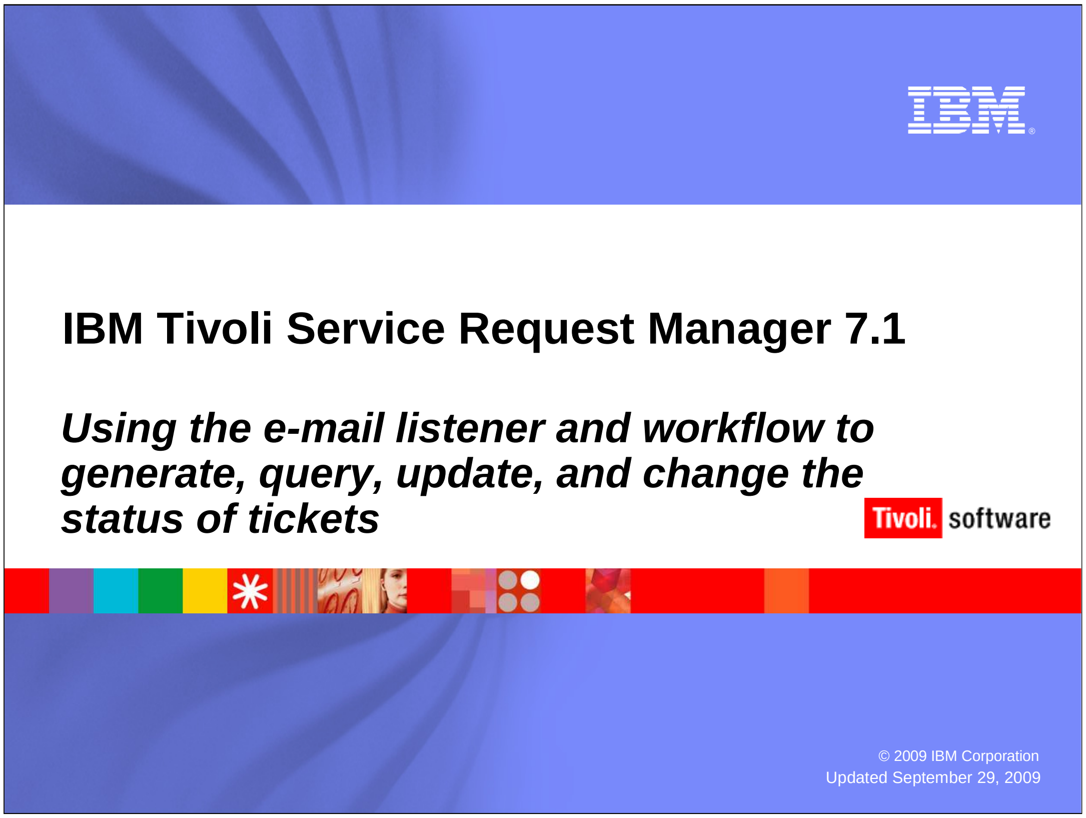


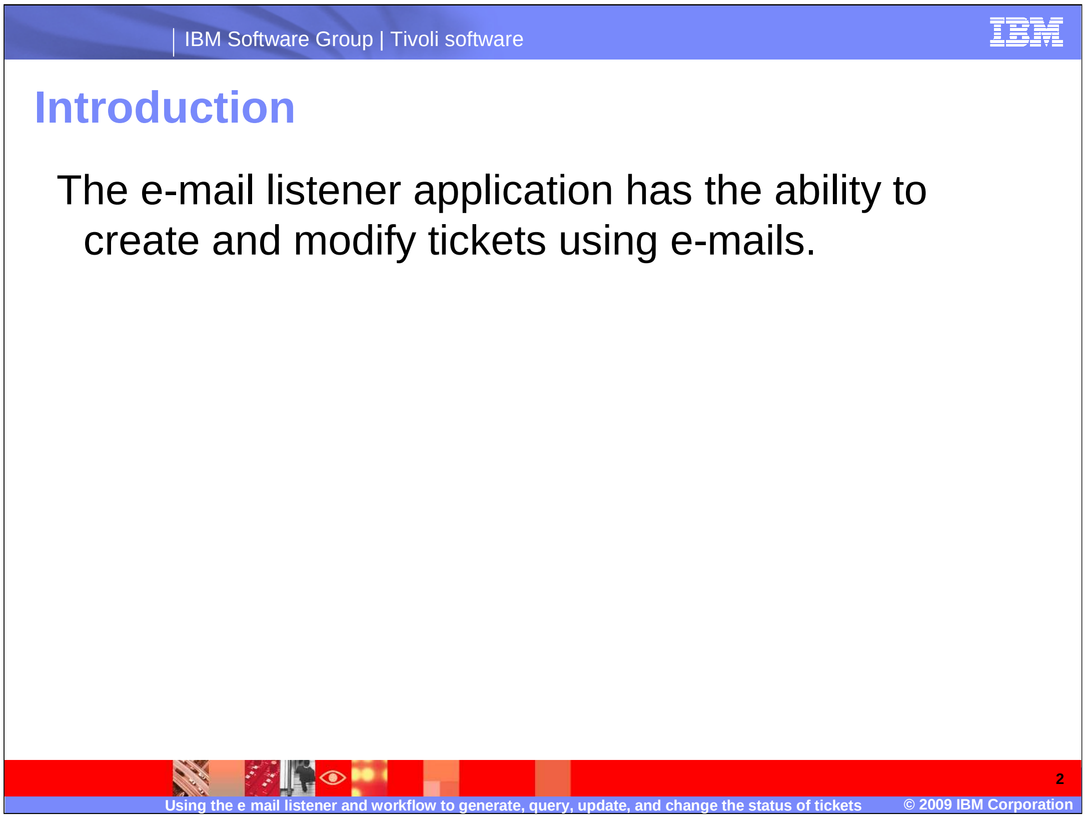
IBM Tivoli Service Request Manager 7.1

Using the e-mail listener and workflow to generate, query, update, and change the status of tickets

Updated September 29, 2009

IBM Tivoli Service Request Manager® 7.1

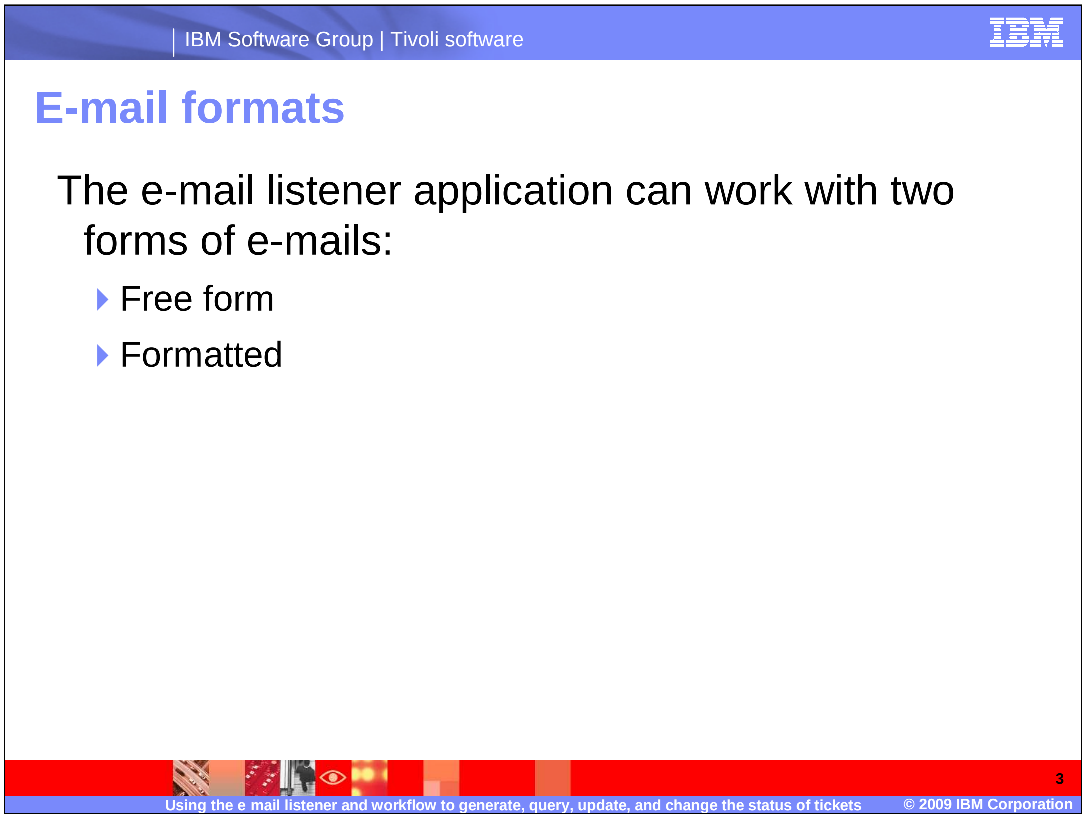
Using the e-mail listener and workflow to generate, query, update, and change the status of tickets



Introduction

The e-mail listener application has the ability to create and modify tickets using e-mails.

Using specially formatted e-mails, the E-mail Listener application (using the LSNRPB workflow) has the ability to create, query, update, and change status of, a ticket.

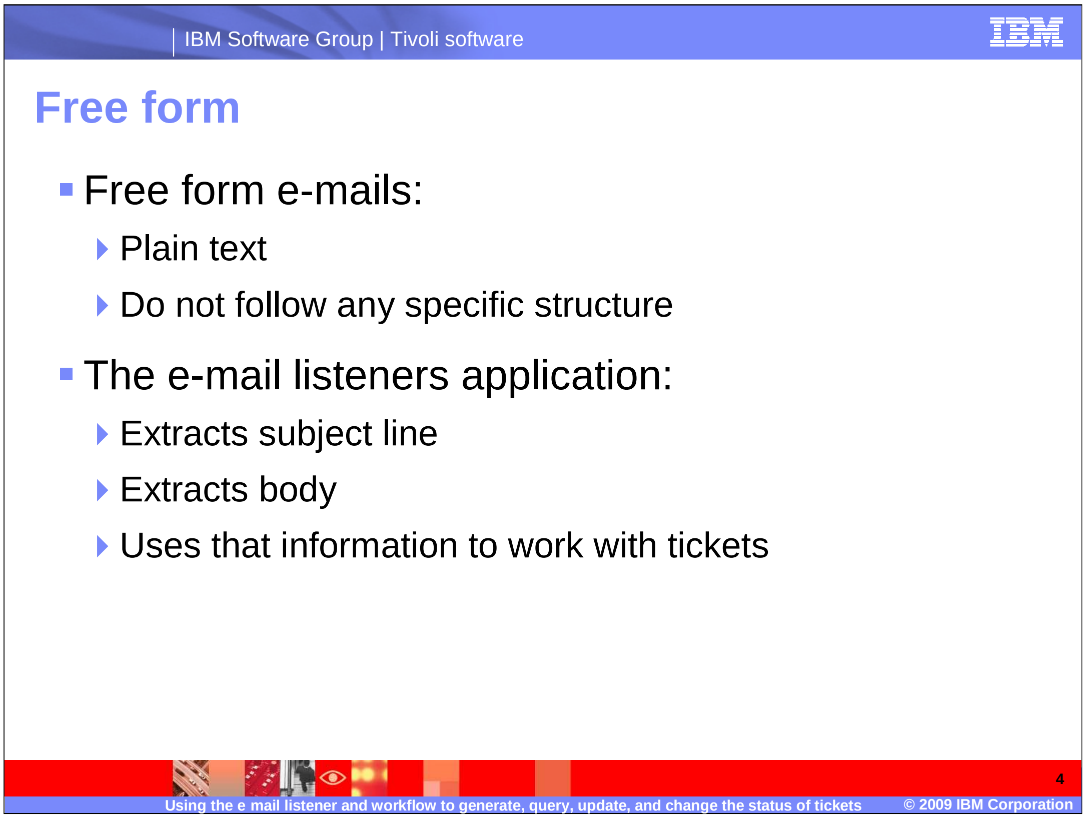


E-mail formats

The e-mail listener application can work with two forms of e-mails:

- Free form
- Formatted

The E-mail Listener application can work with two forms of e-mails: Free Form and Formatted.



IBM Software Group | Tivoli software

Free form

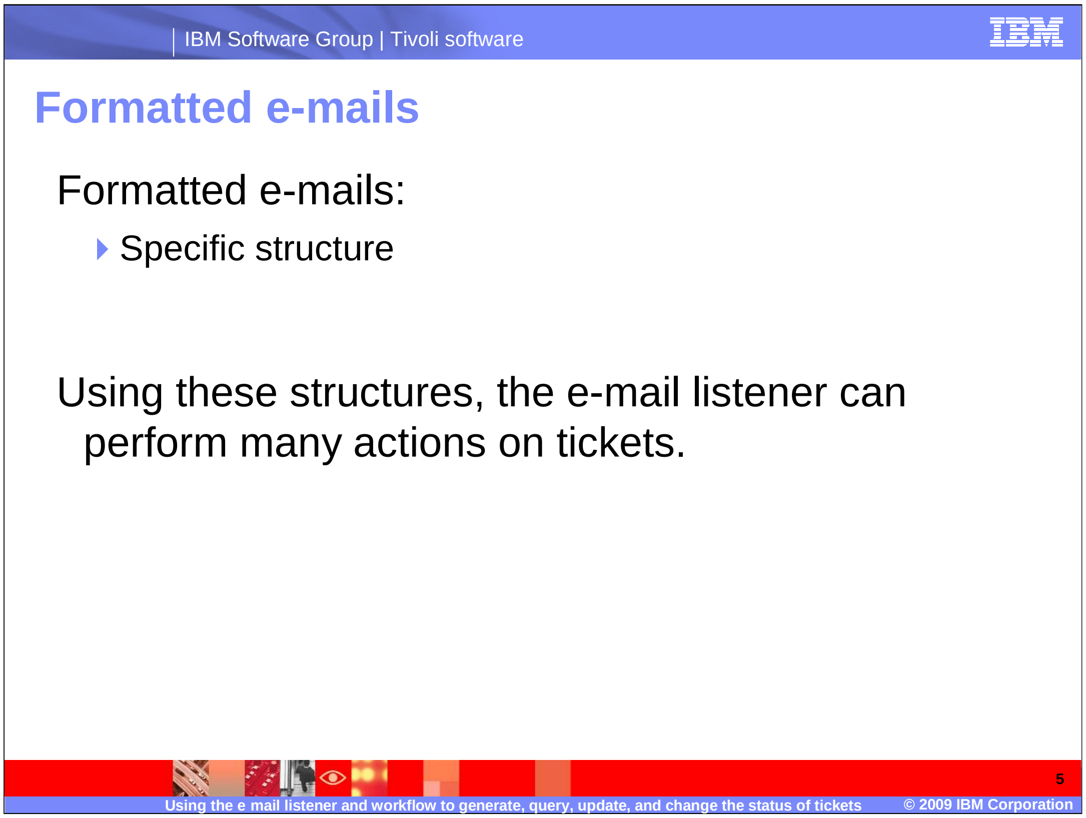
- Free form e-mails:
  - Plain text
  - Do not follow any specific structure
- The e-mail listeners application:
  - Extracts subject line
  - Extracts body
  - Uses that information to work with tickets

4

Using the e mail listener and workflow to generate, query, update, and change the status of tickets

Free form e-mails are in plain text and do not follow any specific structure. The E-mail Listeners application extracts the subject line and body of free form e-mails and uses them to either create a Service Request or update an existing ticket. Free form e-mails are always processed as Service Requests. If the use of other types of tickets is planned, use formatted messages only.

For more details on using free form e-mails, see the *Setting up and using the e-mail listener* training.



Formatted e-mails

Formatted e-mails:

- Specific structure

Using these structures, the e-mail listener can perform many actions on tickets.

Formatted e-mails use a specific structure in the message body.

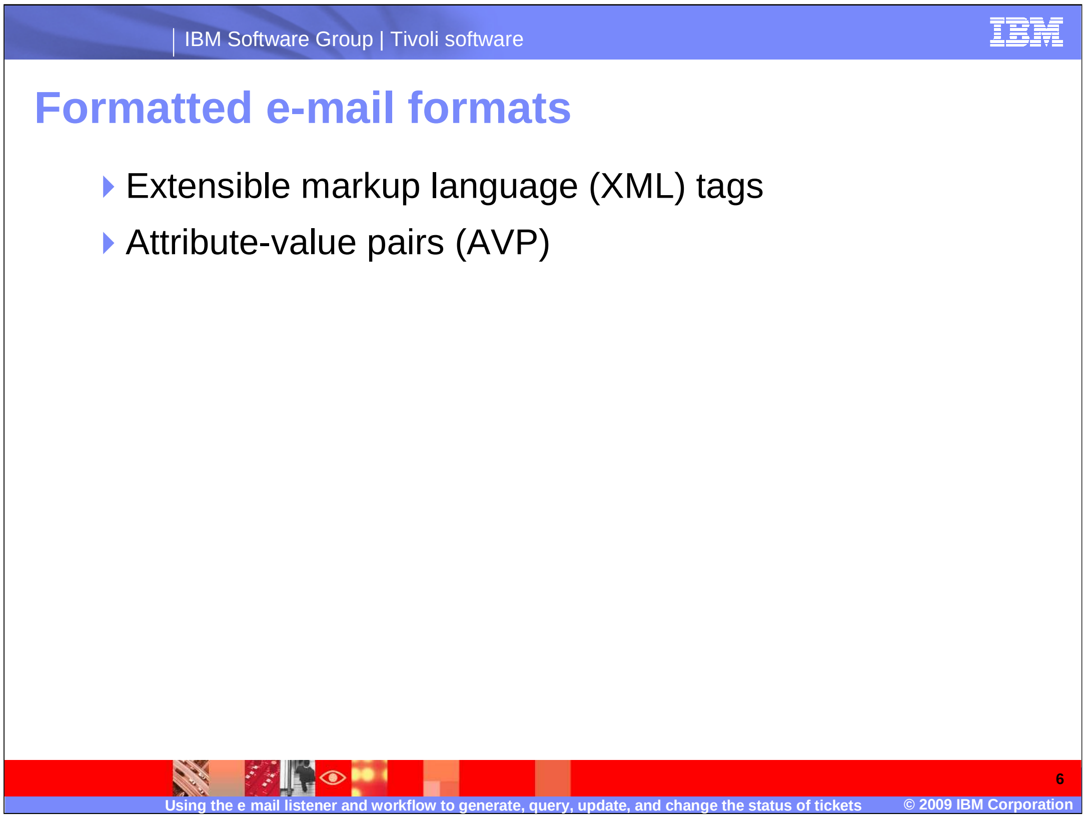
Using these e-mails, the e-mail listener has the ability to perform the following actions:

Create a new ticket

Query an existing ticket

Update ticket information

Change the ticket status



IBM Software Group | Tivoli software

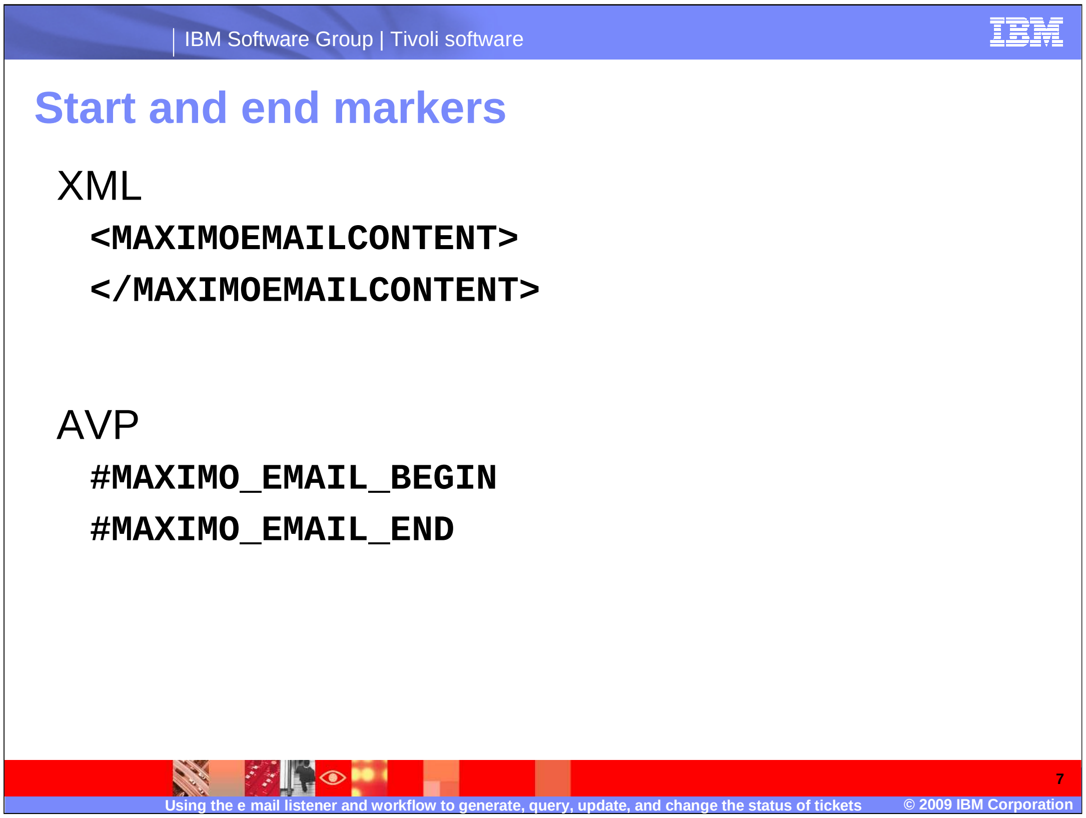
Formatted e-mail formats

- Extensible markup language (XML) tags
- Attribute-value pairs (AVP)

6

Using the e mail listener and workflow to generate, query, update, and change the status of tickets

E-mails formatted for use with the e-mail listener use two different formats, extensible markup language (commonly known as XML) and Attribute-Value Pairs (known as AVP).



Start and end markers

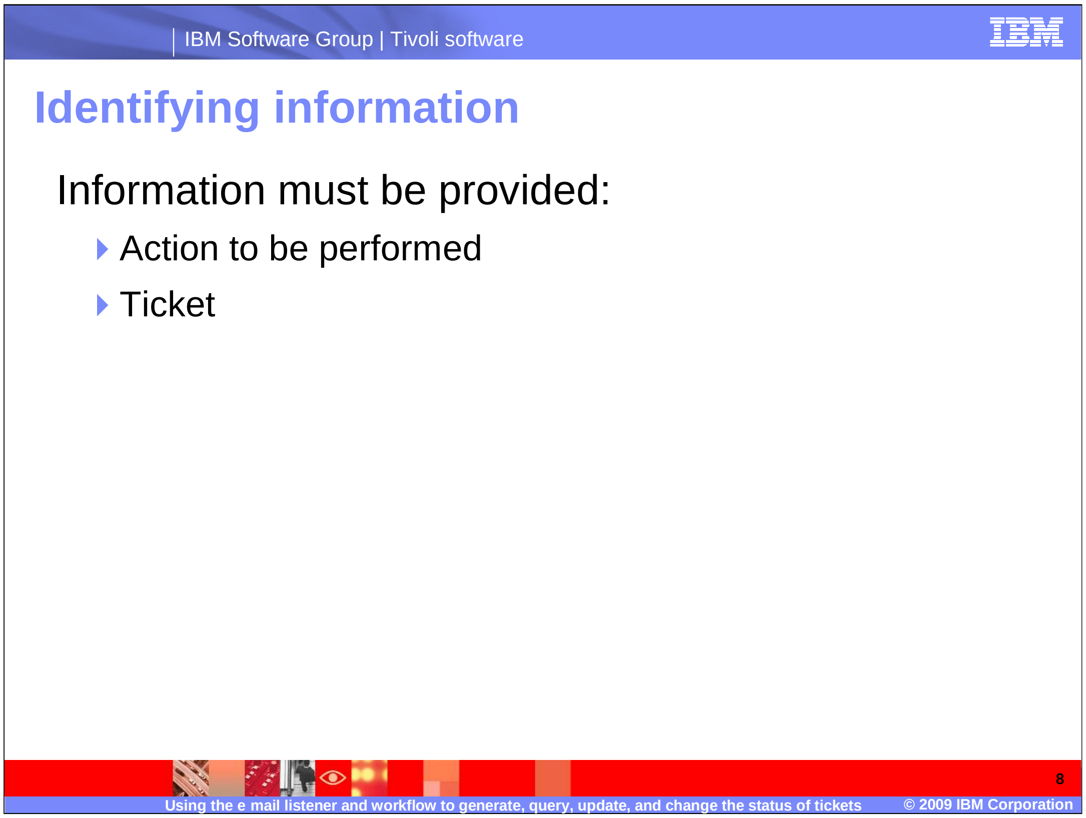
XML

```
<MAXIMOEMAILCONTENT>
</MAXIMOEMAILCONTENT>
```

AVP

```
#MAXIMO_EMAIL_BEGIN
#MAXIMO_EMAIL_END
```

In both the XML and AVP cases, the start and end of the message is signified with keywords.



Identifying information

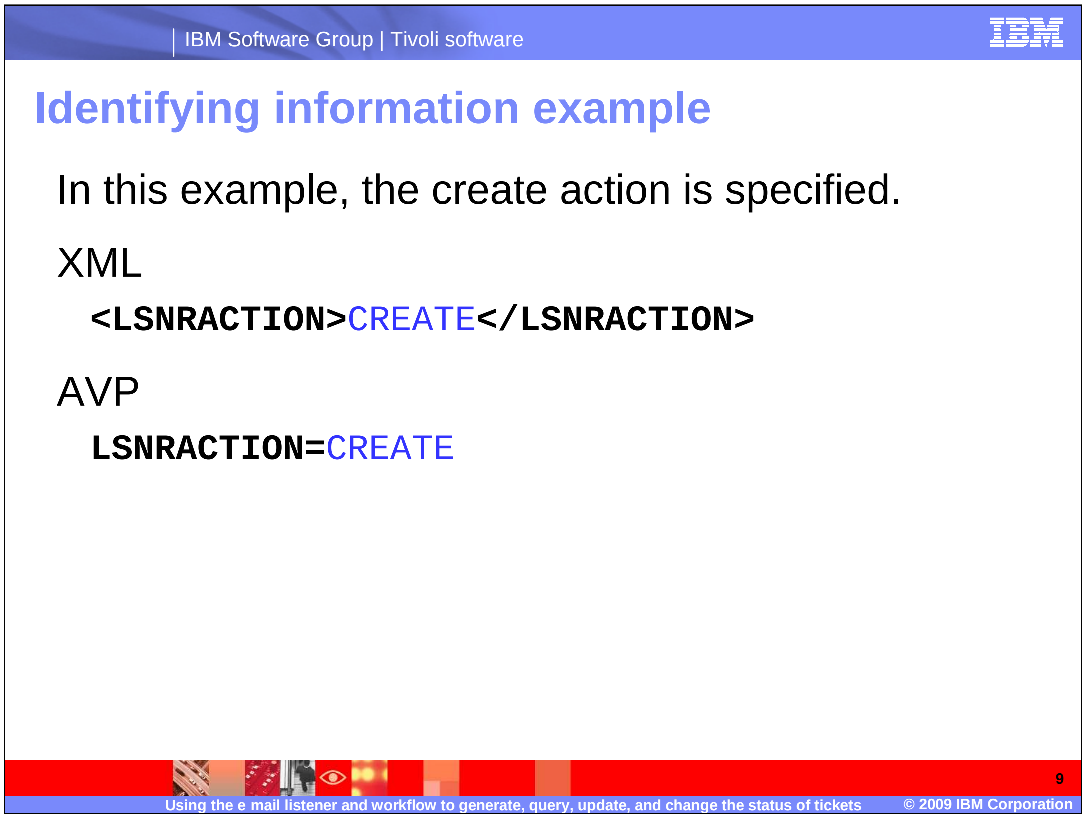
Information must be provided:

- Action to be performed
- Ticket

Information must be provided stating the action to be performed.

In the case of querying, updating, or changing the status of a ticket, information must also be supplied to identify the ticket in question.

This information is placed in between the start and end markers.



Identifying information example

In this example, the create action is specified.

XML

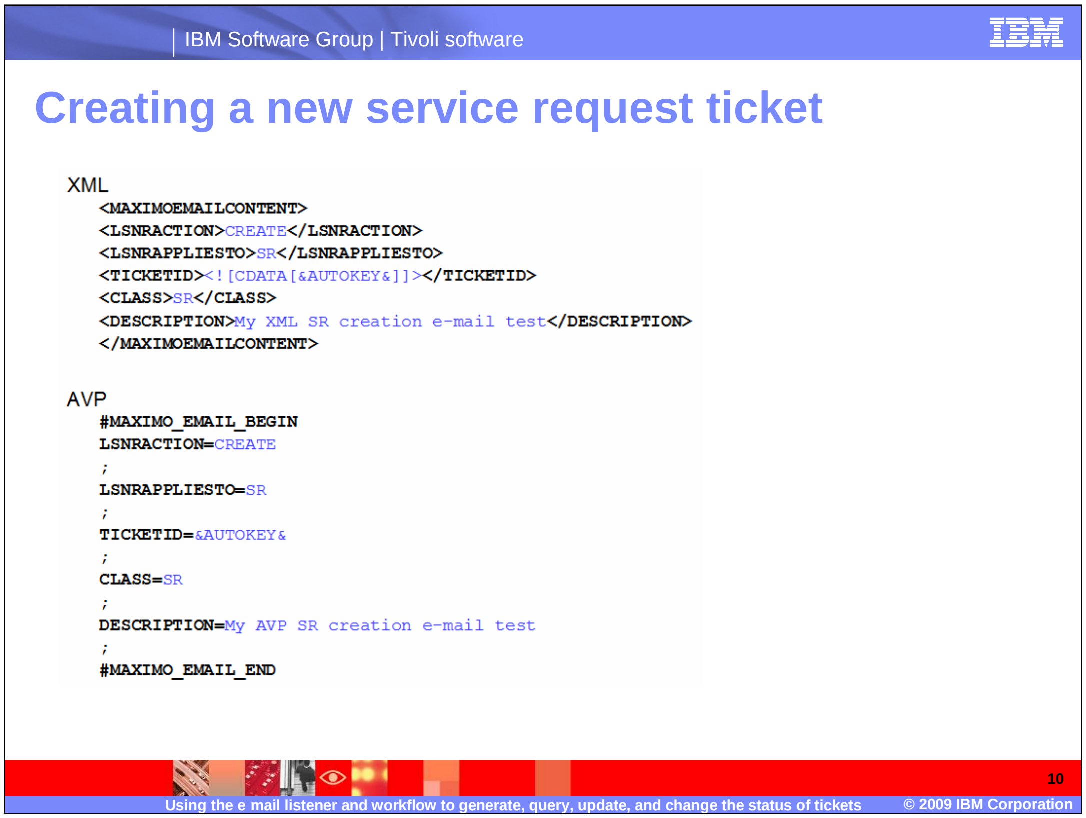
```
<LSNRACTION>CREATE</LSNRACTION>
```

AVP

```
LSNRACTION=CREATE
```

The action to be used must be specified.

Note, for AVP formatted messages a semi-colon character is placed between all attribute value groups. For XML formatted messages, if the keywords &AUTOKEY& or &SYSDATE& must be used, these keywords must be escaped using standard XML CDATA constructs.



IBM Software Group | Tivoli software

Creating a new service request ticket

XML

```
<MAXIMOEMAILCONTENT>
<LSNRACTION>CREATE</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<TICKETID><![CDATA[&AUTOKEY&]]></TICKETID>
<CLASS>SR</CLASS>
<DESCRIPTION>My XML SR creation e-mail test</DESCRIPTION>
</MAXIMOEMAILCONTENT>
```

AVP

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=CREATE
;
LSNRAPPLIESTO=SR
;
TICKETID=&AUTOKEY&
;
CLASS=SR
;
DESCRIPTION=My AVP SR creation e-mail test
;
#MAXIMO_EMAIL_END
```

10

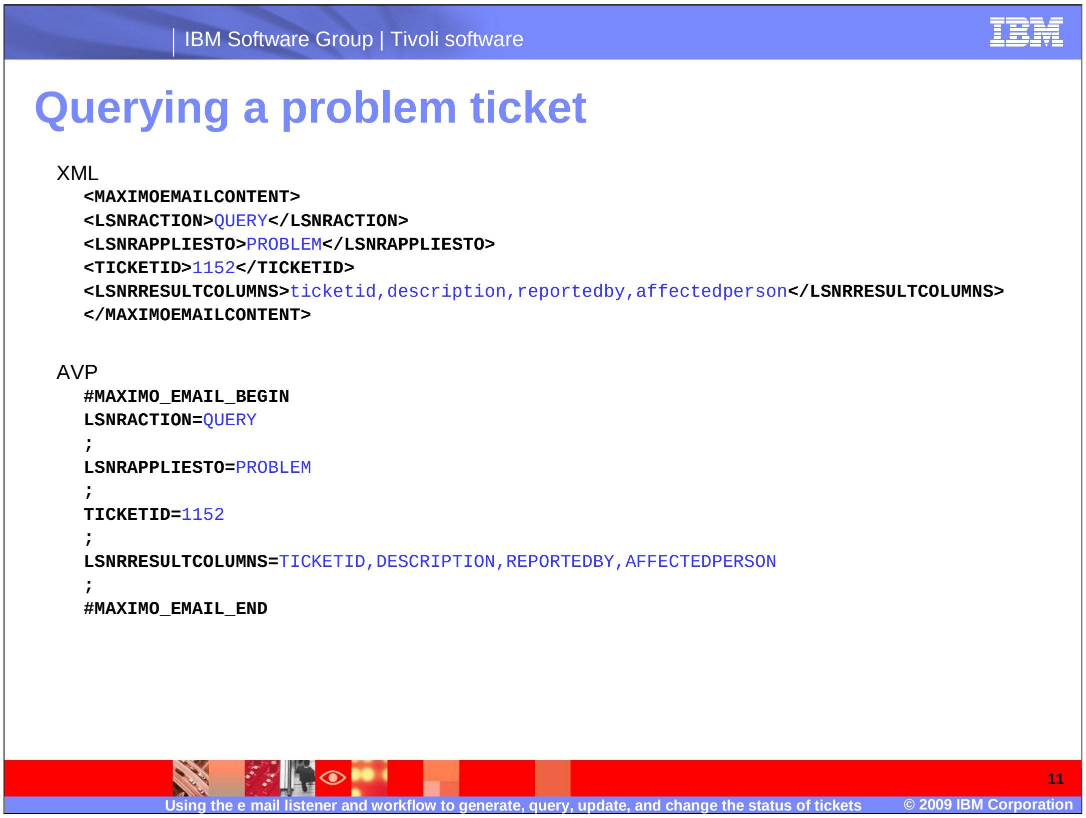
Using the e mail listener and workflow to generate, query, update, and change the status of tickets

Starting with this slide, examples of the different actions that can be performed on tickets will be shown.

This example covers creating a new service request using a formatted message.

There are two different examples here, XML and AVP. They both specify the *action type* of *create*, the *applies to* type of *SR* for service request, the *ticket ID* of *Autokey*, a *class type* of *SR*, description of *My XML or AVP SR creation e-mail test* and finally the close tag.

Note that in the XML version, the &AUTOKEY& information has been wrapped in a CDATA construct.



Querying a problem ticket

XML

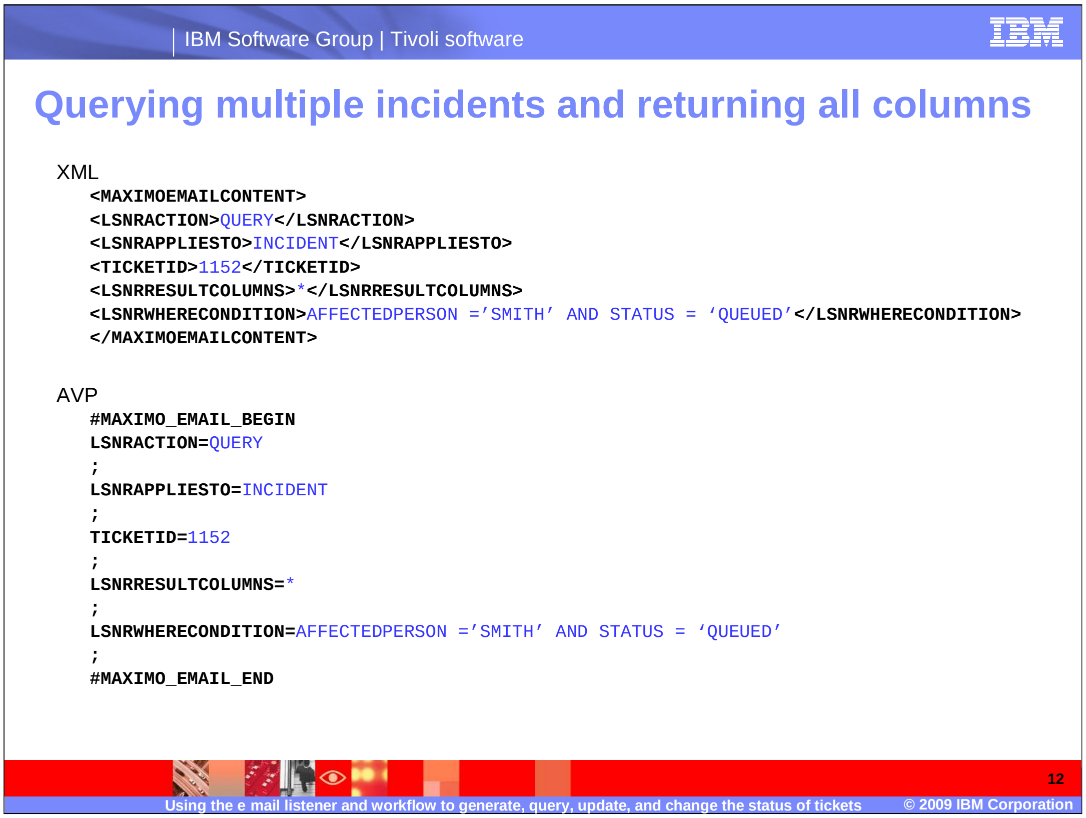
```
<MAXIMOEMAILCONTENT>
<LSNRACTION>QUERY</LSNRACTION>
<LSNRAPPLIESTO>PROBLEM</LSNRAPPLIESTO>
<TICKETID>1152</TICKETID>
<LSNRRESULTCOLUMNS>ticketid,description,reportedby,affectedperson</LSNRRESULTCOLUMNS>
</MAXIMOEMAILCONTENT>
```

AVP

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=QUERY
;
LSNRAPPLIESTO=PROBLEM
;
TICKETID=1152
;
LSNRRESULTCOLUMNS=TICKETID,DESCRIPTION,REPORTEDBY,AFFECTEDPERSON
;
#MAXIMO_EMAIL_END
```

In this example covering the querying of a problem ticket, specified are an action type of *query*, applies to of *problem*, the *ticket id*, the columns to be returned, in this case *ticket id*, *description*, *reported by*, and *affected person*, and finally the close tags.

Note that each column to be returned is specified.



Querying multiple incidents and returning all columns

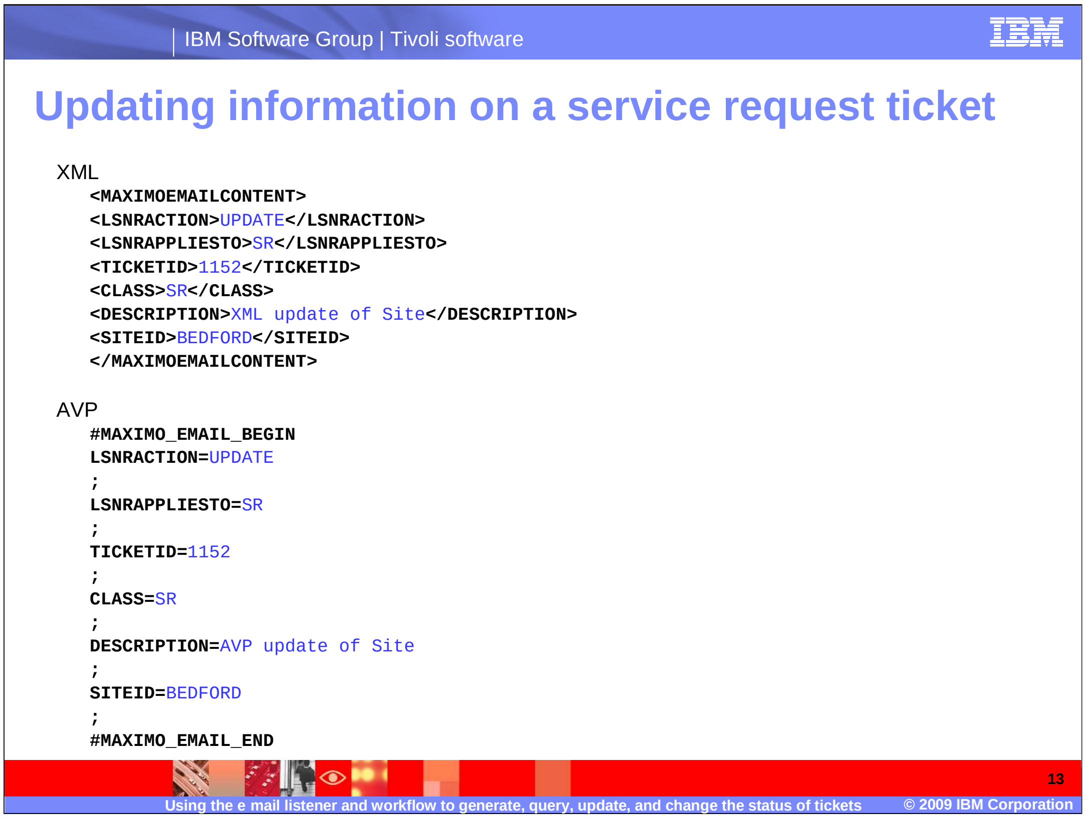
XML

```
<MAXIMOEMAILCONTENT>
<LSNRACTION>QUERY</LSNRACTION>
<LSNRAPPLIESTO>INCIDENT</LSNRAPPLIESTO>
<TICKETID>1152</TICKETID>
<LSNRRESULTCOLUMNS>*</LSNRRESULTCOLUMNS>
<LSNRWHERECONDITION>AFFECTEDPERSON =’SMITH’ AND STATUS = ‘QUEUED’</LSNRWHERECONDITION>
</MAXIMOEMAILCONTENT>
```

AVP

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=QUERY
;
LSNRAPPLIESTO=INCIDENT
;
TICKETID=1152
;
LSNRRESULTCOLUMNS=*
;
LSNRWHERECONDITION=AFFECTEDPERSON =’SMITH’ AND STATUS = ‘QUEUED’
;
#MAXIMO_EMAIL_END
```

These examples are the same as the previous example, except for the difference of, to return all columns the asterisk symbol is used. A where condition has been added that allows for the selection of the person and status of the ticket.



IBM Software Group | Tivoli software

Updating information on a service request ticket

XML

```
<MAXIMOEMAILCONTENT>
<LSNRACTION>UPDATE</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<TICKETID>1152</TICKETID>
<CLASS>SR</CLASS>
<DESCRIPTION>XML update of Site</DESCRIPTION>
<SITEID>BEDFORD</SITEID>
</MAXIMOEMAILCONTENT>
```

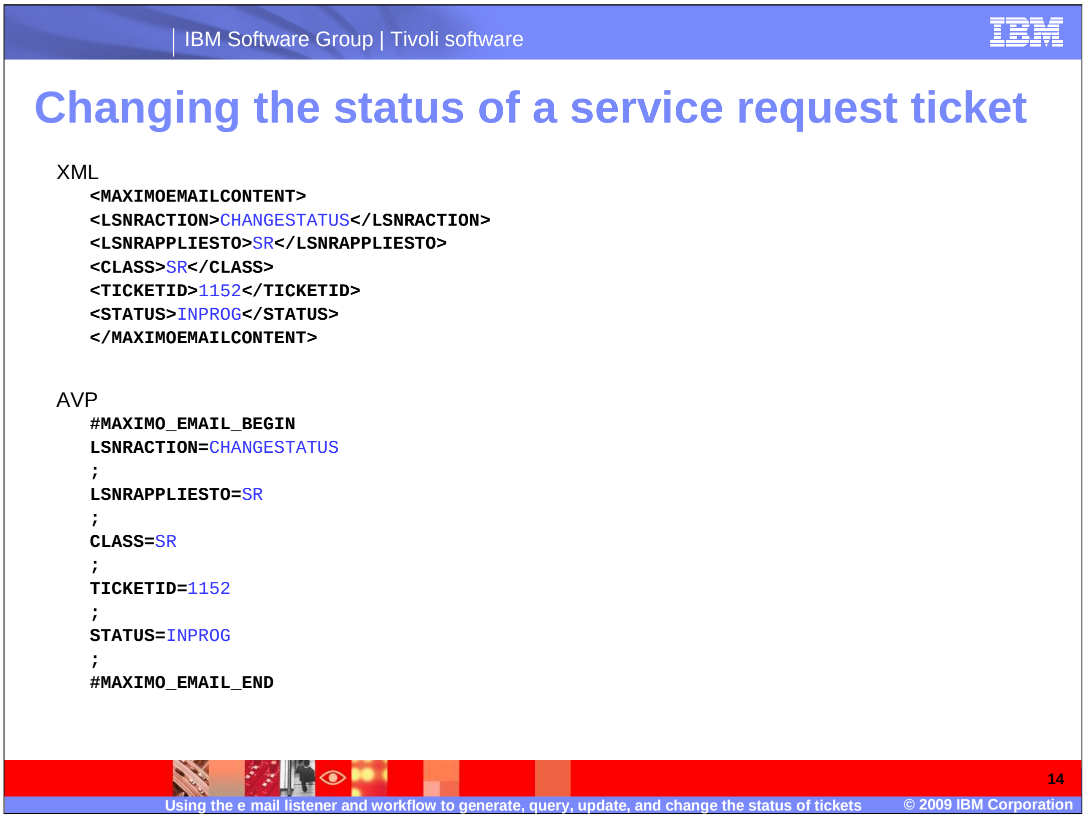
AVP

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=UPDATE
;
LSNRAPPLIESTO=SR
;
TICKETID=1152
;
CLASS=SR
;
DESCRIPTION=AVP update of Site
;
SITEID=BEDFORD
;
#MAXIMO_EMAIL_END
```

13

Using the e mail listener and workflow to generate, query, update, and change the status of tickets

In this example covering the updating of information on a ticket, an *action* type of *update*, an *applies to* type of SR, a *ticketid*, *class*, *description* and *siteid* are specified.



IBM Software Group | Tivoli software

Changing the status of a service request ticket

XML

```
<MAXIMOEMAILCONTENT>
<LSNRACTION>CHANGESTATUS</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<CLASS>SR</CLASS>
<TICKETID>1152</TICKETID>
<STATUS>INPROG</STATUS>
</MAXIMOEMAILCONTENT>
```

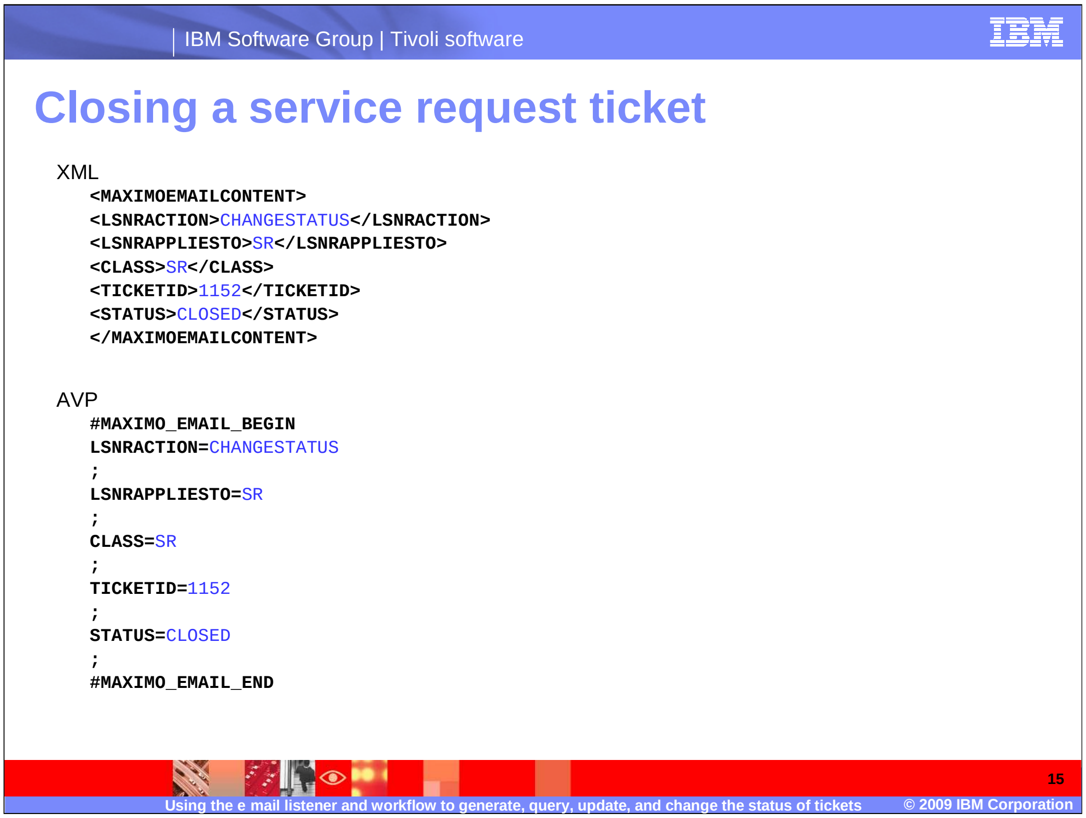
AVP

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=CHANGESTATUS
;
LSNRAPPLIESTO=SR
;
CLASS=SR
;
TICKETID=1152
;
STATUS=INPROG
;
#MAXIMO_EMAIL_END
```

14

Using the e mail listener and workflow to generate, query, update, and change the status of tickets

In this example of changing the status of a ticket, an *action* type of *change status*, an *applies to* type of SR, a *ticket id*, *class*, and *status* are specified.



Closing a service request ticket

XML

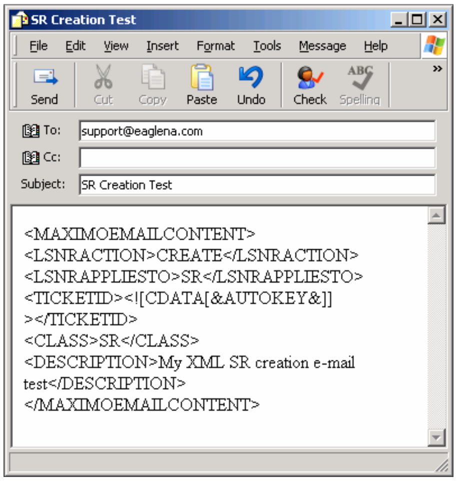
```
<MAXIMOEMAILCONTENT>
<LSNRACTION>CHANGESTATUS</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<CLASS>SR</CLASS>
<TICKETID>1152</TICKETID>
<STATUS>CLOSED</STATUS>
</MAXIMOEMAILCONTENT>
```

AVP

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=CHANGESTATUS
;
LSNRAPPLIESTO=SR
;
CLASS=SR
;
TICKETID=1152
;
STATUS=CLOSED
;
#MAXIMO_EMAIL_END
```

Closing a ticket only takes a simple status change, so it is given an *action* type of *change status*, an *applies to* type of SR, a *ticket id*, *class*, and a *status* of *closed.*

Create service request outgoing e-mail



To: support@eaglena.com

Cc:

Subject: SR Creation Test

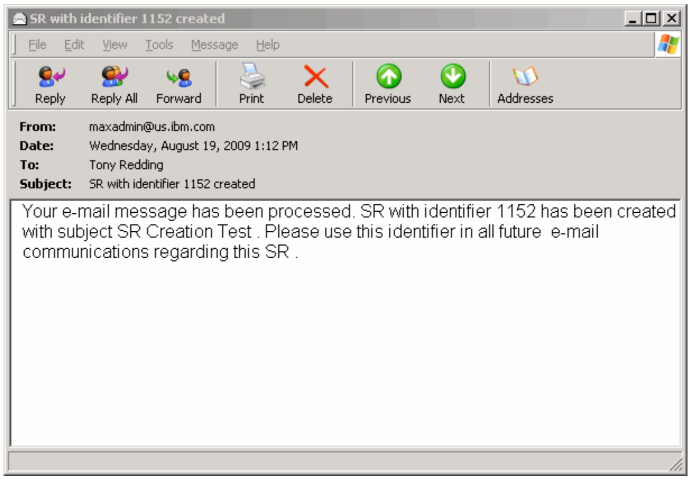
```
<MAXIMOEMAILCONTENT>
<LSNRACTION>CREATE</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<TICKETID><![CDATA[&AUTOKEY&]]
></TICKETID>
<CLASS>SR</CLASS>
<DESCRIPTION>My XML SR creation e-mail
test</DESCRIPTION>
</MAXIMOEMAILCONTENT>
```

Now, real world examples of formatted e-mails being sent to and received from the e-mail listener will be covered.

This example will show a service request created using an outgoing e-mail.

First, an XML formatted message is sent asking for the creation of a new service request.

Create service request incoming e-mail

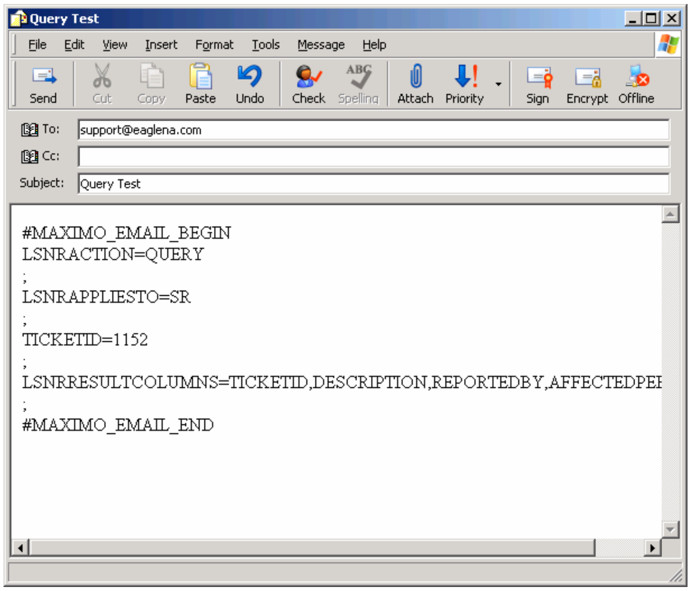

SR with identifier 1152 created
File Edit View Tools Message Help
Reply
Reply All
Forward
Print
Delete
Previous
Next
Addresses

From: maxadmin@us.ibm.com
Date: Wednesday, August 19, 2009 1:12 PM
To: Tony Redding
Subject: SR with identifier 1152 created

Your e-mail message has been processed. SR with identifier 1152 has been created with subject SR Creation Test . Please use this identifier in all future e-mail communications regarding this SR .

And here is the reply sent from the e-mail listener verifying that the service request was created.

Query service request outgoing e-mail



To: support@eaglena.com

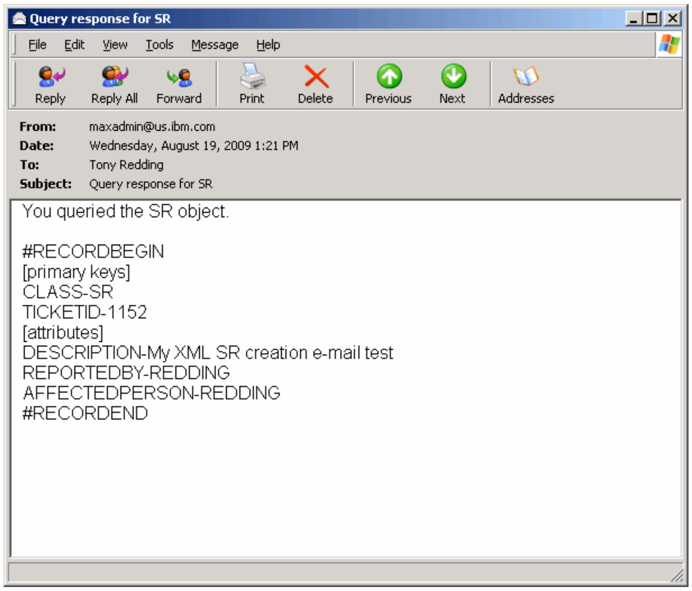
Cc:

Subject: Query Test

```
#MAXIMO_EMAIL_BEGIN
LSNRACTION=QUERY
;
LSNRAPPLIESTO=SR
;
TICKETID=1152
;
LSNRRESULTCOLUMNS=TICKETID,DESCRIPTION,REPORTEDBY,AFFECTEDPE
;
#MAXIMO_EMAIL_END
```

In the next example, an AVP formatted e-mail is sent querying a few fields from the newly created service request.

Query service request incoming e-mail



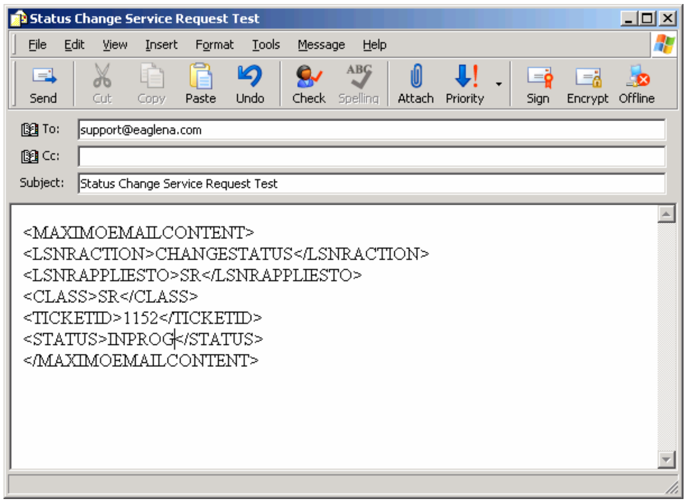
From: maxadmin@us.ibm.com
Date: Wednesday, August 19, 2009 1:21 PM
To: Tony Redding
Subject: Query response for SR

```
You queried the SR object.

#RECORDBEGIN
[primary keys]
CLASS-SR
TICKETID-1152
[attributes]
DESCRIPTION-My XML SR creation e-mail test
REPORTEDBY-REDDING
AFFECTEDPERSON-REDDING
#RECORDEND
```

And the reply giving the requested information is received.

Service request status change outgoing e-mail


Status Change Service Request Test
File Edit View Insert Format Tools Message Help
Send
Cut
Copy
Paste
Undo
Check
Spelling
Attach
Priority
Sign
Encrypt
Offline

To: support@eaglena.com

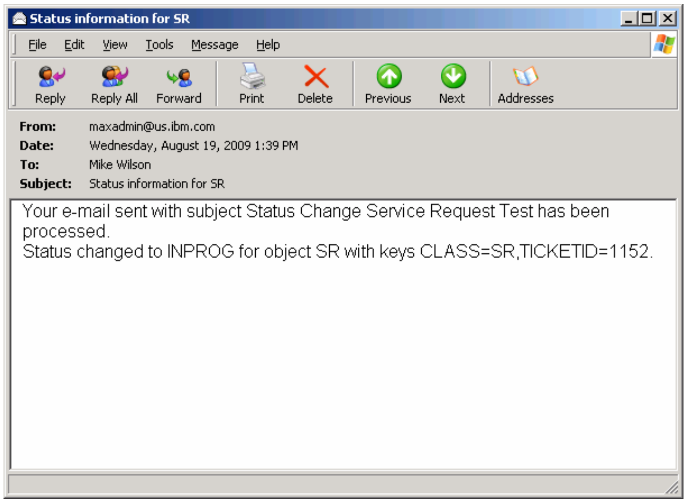
Cc:

Subject: Status Change Service Request Test

```
<MAXIMOEMAILCONTENT>
<LSNRACTION>CHANGESTATUS</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<CLASS>SR</CLASS>
<TICKETID>1152</TICKETID>
<STATUS>INPROG</STATUS>
</MAXIMOEMAILCONTENT>
```

In this next example, an XML formatted message is requesting the change of a Service Requests’ status to *In Progress*.

Service request status change incoming e-mail

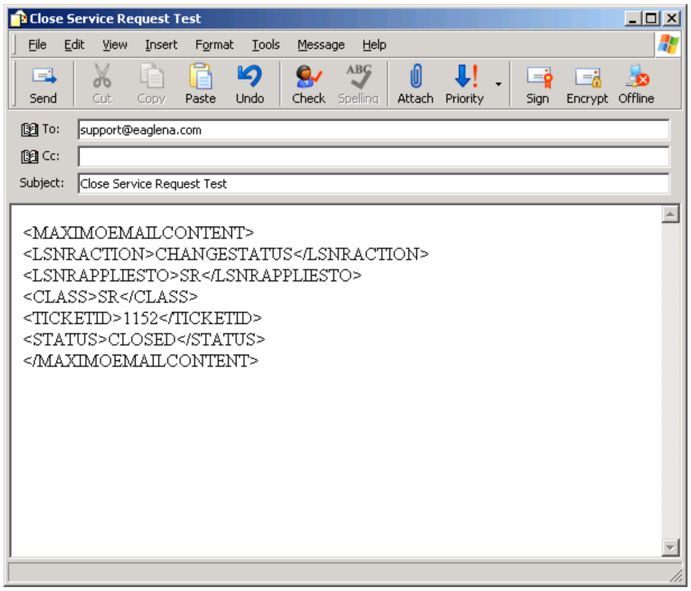


From: maxadmin@us.ibm.com
Date: Wednesday, August 19, 2009 1:39 PM
To: Mike Wilson
Subject: Status information for SR

Your e-mail sent with subject Status Change Service Request Test has been processed.
Status changed to INPROG for object SR with keys CLASS=SR,TICKETID=1152.

And the reply comes back confirming the successful change.

Close service request outgoing e-mail (unauthorized)


Close Service Request Test
File Edit View Insert Format Tools Message Help
Send
Cut
Copy
Paste
Undo
Check
Spelling
Attach
Priority
Sign
Encrypt
Offline

To: support@eaglena.com

Cc:

Subject: Close Service Request Test

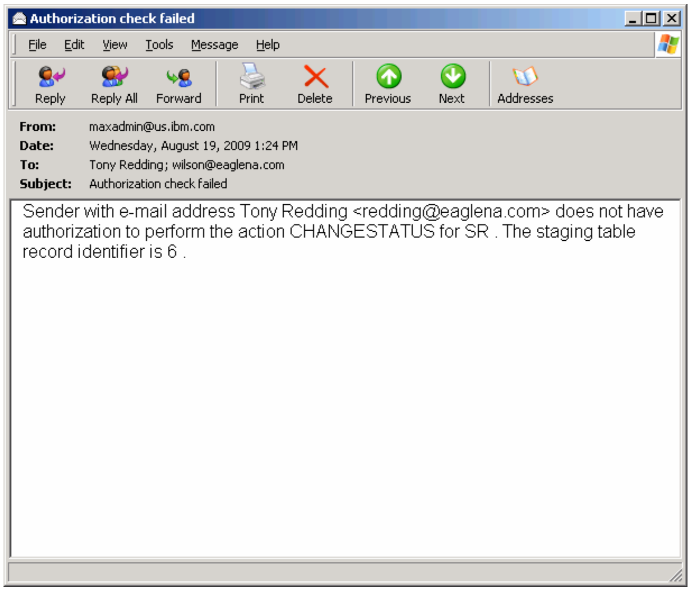
```
<MAXIMOEMAILCONTENT>
<LSNRACTION>CHANGESTATUS</LSNRACTION>
<LSNRAPPLIESTO>SR</LSNRAPPLIESTO>
<CLASS>SR</CLASS>
<TICKETID>1152</TICKETID>
<STATUS>CLOSED</STATUS>
</MAXIMOEMAILCONTENT>
```

This final example deals with an unauthorized person requesting the closure of a ticket.

Not everyone has the proper permissions to change ticket information. If they would not have permission to change the information from inside of Service Request Manager, then they will not be able to change it using the e-mail listener.

Although it is not obvious from the image, this close request is being sent from the account of a user who is not authorized to change the status of a ticket.

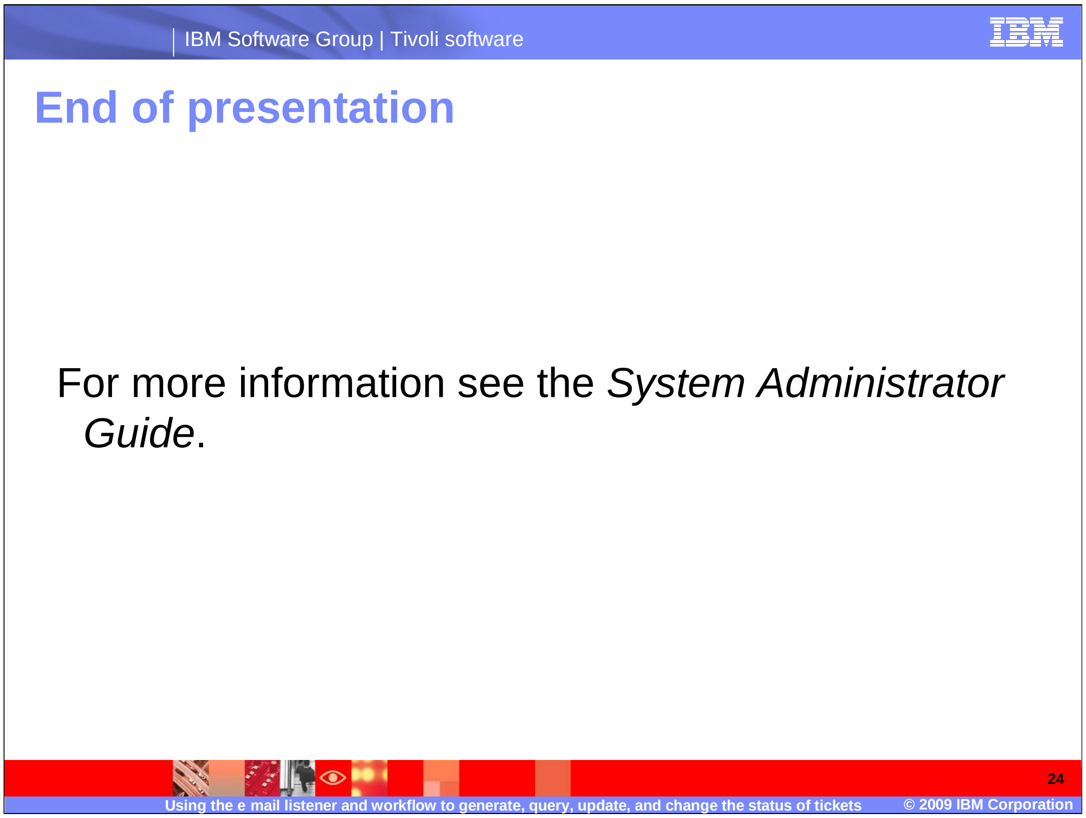
Close service request incoming e-mail (unauthorized)



From: maxadmin@us.ibm.com
Date: Wednesday, August 19, 2009 1:24 PM
To: Tony Redding; wilson@eaglena.com
Subject: Authorization check failed

Sender with e-mail address Tony Redding <redding@eaglena.com> does not have authorization to perform the action CHANGESTATUS for SR . The staging table record identifier is 6 .

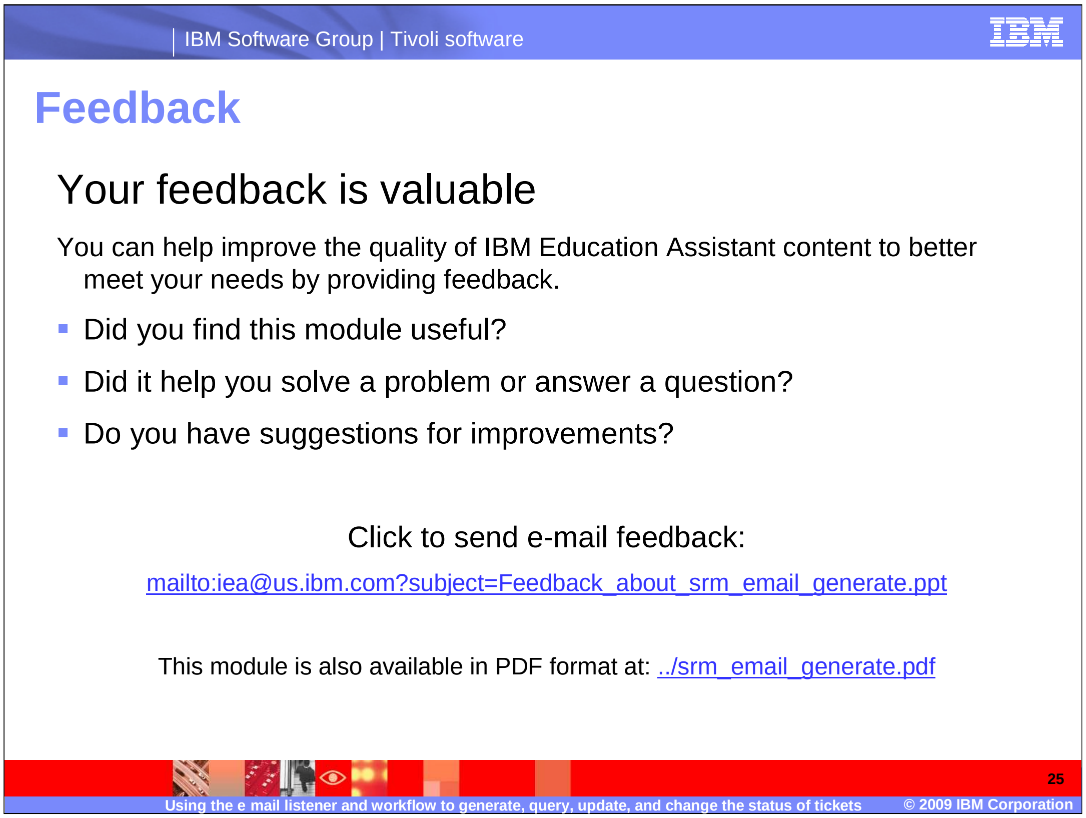
And the reply is received stating that the user is not authorized to change the status of the service request.



End of presentation

For more information see the *System Administrator Guide*.

For more information see the *IBM Tivoli Service Request Manager 7.1 System Administrator Guide*.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_srm_email_generate.ppt

This module is also available in PDF format at: ../srm_email_generate.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	Service Request Manager	Tivoli

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

Other company, product, or service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.