



# Whitepaper

## **CORE for Anti-Spam**

- Innovative Spam Protection -

Mastering the challenge of spam today with the technology  
of tomorrow

*Expertise matters*

## Contents

1	Spam Defense – An Overview .....	3
1.1	Efficient Spam Protection – Procedure .....	4
2	Potential Applications of CORE.....	6
3	How Does the CORE Technology Work? .....	7
3.1	SVM Basics .....	7
3.2	Why SVM? .....	8
4	Practical Use of CORE for Anti-Spam.....	10
4.1	Text Analysis with CORE .....	10
4.2	Practical Tips.....	11
4.2.1	Categorization.....	11
4.2.2	Training and Validation .....	11
4.2.3	Recategorization .....	12
5	CORE for Anti-Spam – Highlights and Features.....	13

# 1 Spam Defense – An Overview

SPAM – everybody knows it, nobody likes it, and absolutely no one wants it.

Protecting against spam is an ongoing technological challenge. Continuous changes in technology and the imagination of spammers mean that this challenge constantly requires new defensive measures. What are the opportunities for detecting this electronic garbage even before it lands in the recipient's inbox?

Defensive measures against spam range from extremely simple string operations to complex script-based programs. Overall, they can be divided into the following methods:

- Word lists (dictionaries) are one of the oldest methods for protection against spam. As the risk of false positives (= desirable email classified as spam) is quite high, this method is unsuitable as a primary defense mechanism.
- Real-time blacklists (RBLs or DNSBLs) are a method in which blacklist servers available on the Internet are queried to identify an email sender as a spammer. This method is extremely insecure, because email servers of serious senders frequently turn up on such lists following a spam attack. Moreover, the address lists stored on these blacklist servers rapidly become outdated. The error rate of RBLs (i.e. false positives and unrecognized spam mails) is already at a very high rate of 60% or more.
- Checksum methods create a unique checksum for each incoming email, and store these in Internet databases once they have been categorized. Other email servers can compare incoming emails with this database and detect email that has been classified as spam. Some existing solutions offer a service that provides current updates/"patterns" for spam detection, similar to antivirus programs. The checksum method is based on the assumption that spam emails may be replicated copies of the same email, and can therefore be clearly allocated to all recipient servers. In order to function, this procedure relies upon the largest possible network of participants. This procedure has since been frequently circumvented by spammers, however, by making the generated spam email distinguishable only in the checksum but not in the legible text, or by sending mass emails as personalized individual email messages.
- HTML decoding addresses the fact that spammers are increasingly sending HTML-based emails in order to circumvent standard spam detection methods. In doing so, spammers use the tagging possibilities of HTML in a form that allows the client to display a legible email, but which contains no contiguous text in the HTML source code. The HTML decoding methods decode the HTML-based email and check a variety of elements, from typical formatting such as capital letters and colors to integrated HTML links.

- Script filters represent an effective, although highly complex and expensive method of using Perl or Sieve scripts to achieve a customized spam defense. Higher development costs as well as ongoing maintenance expenses make methods in this category very inefficient and difficult to manage, however.
- Heuristic approaches attempt to detect certain text patterns in emails that may permit them to be classified as spam or non-spam.

In addition to various approaches using neuronal networks and Bayes filters, the innovative CORE (Content Recognition Engine) analysis method belongs to this category.

The naïve Bayes filters are a frequently used method at present. This statistical method can be traced back to an 18th-century minister named Thomas Bayes, who developed the underlying probability formula 250 years ago. Using calculated probabilities, new emails are classified as spam or non-spam. Naïve Bayes filters make no connection between individual document features. This makes them unsuitable for multiple categorization without implementing dedicated add-on modules.

CORE, on the other hand, eliminates both the problems of traditional methods, such as slow and expensive adaptation to new spamming techniques, and the generally high training expense of other heuristic methods.

## 1.1 Efficient Spam Protection – Procedure

Various methods are combined to produce an efficient and powerful defense against spam. The following sequence has proven effective in practice for email analysis:

1. Check address using blacklists (prohibited email addresses and domains) and company-specific whitelists (permitted addresses and domains). The whitelists contain business-relevant sender addresses, e.g. for customers, suppliers, newsletters, discussion forums.
2. Check subject line for simple keywords using dictionary (100% stop words).
3. Check message text using dictionary and HTML analysis. In this step, the dictionaries used should contain 100% stop words, similar to the dictionary for the subject line, in order to immediately sort out corresponding emails as spam. Dictionaries with 100% stop words are generally shorter and require less maintenance.
4. Check email content with CORE.

### SPAM-free in 4 Steps:

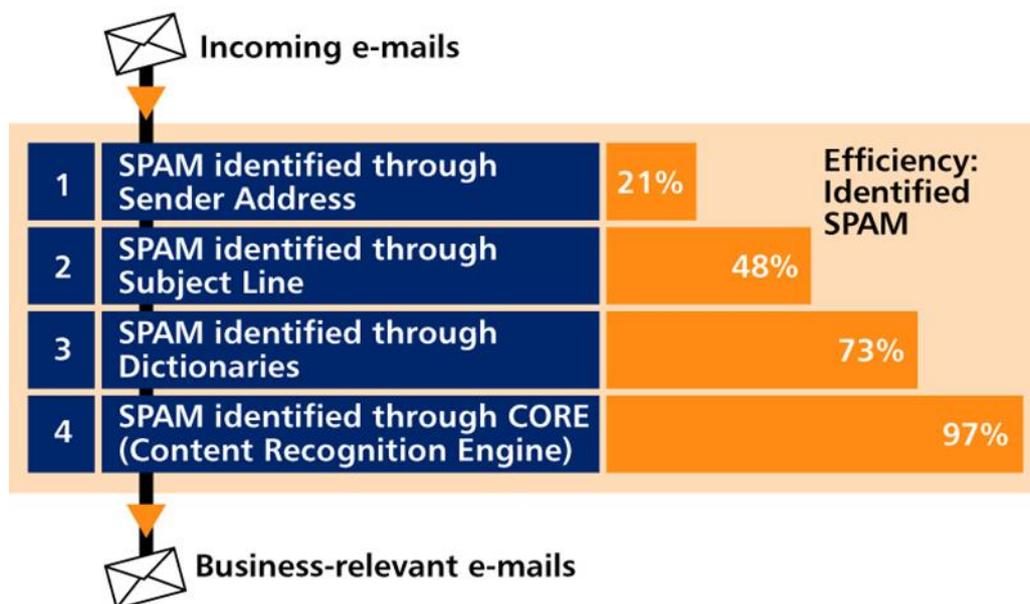


Illustration 1: Spam defense with iQ.Suite Wall

Without CORE, a user can eliminate no more than 73% of spam (see fig.) – and this is a declining trend, because new spammer tricks can circumvent these static methods. With CORE, however, a user can detect 97% of incoming spam email now and in the future, because CORE is an adaptive technique and learns to recognize new spammer techniques as they appear.

CORE anti-spam features:

- Spam detection rate above 95%
- Reduction of false positives to less than 0.1% within emails identified as spam.
- Simple categorization into two categories only: SPAM and NOSPAM.

Multiple categorization to distinguish between spam and business email: spam, newsletters, quotations, orders, private mail, etc. Newsletters and spam, for instance, often have a similar structure and may also have relatively similar content. CORE uses multiple categories to permit the most precise classification of these emails.

## 2 Potential Applications of CORE

CORE analyzes and classifies email. As an adaptive program, it can be used for the following practical applications:

- **Anti-spam:** Spammers are constantly refining their methods in order to slip by static filters that analyze keywords, message texts, or subject lines. For example, words or sentences are manipulated in such a way that they will not appear in any dictionaries, but can be correctly displayed by the user's browser or email client, or can be understood by a human reader. Methods for obscuring text include adding characters between letters, e.g. "Burn F\_a\_t" or "H:A:R:D:C:O:R:E E:X:T:R:E:M:E." An increasingly frequent type of spam email has a very business-like or apparently personal message text, and is impossible to detect using traditional methods because it cannot be distinguished from normal correspondence by means of lexical analysis. All of these deceptive spamming tactics are useless against CORE, which recognizes them for what they are, namely spam. CORE does not restrict itself to individual words or sentence fragments, but instead analyzes the entire content of the email. Because of its adaptive capability, CORE also learns to recognize new spammer techniques that may appear in the future.
- **Document protection:** CORE provides better protection for documents that contain internal company information. With a corresponding category set up, for example, all outgoing emails, including attachments, can be checked for company-relevant content.
- **Email response management:** Another possible application of CORE is to optimize internal workflows through email response management. By using preset categories, for example, email sent to customer support can be automatically detected, classified, and sent to the responsible employee, or else answered directly with an automatically generated response.

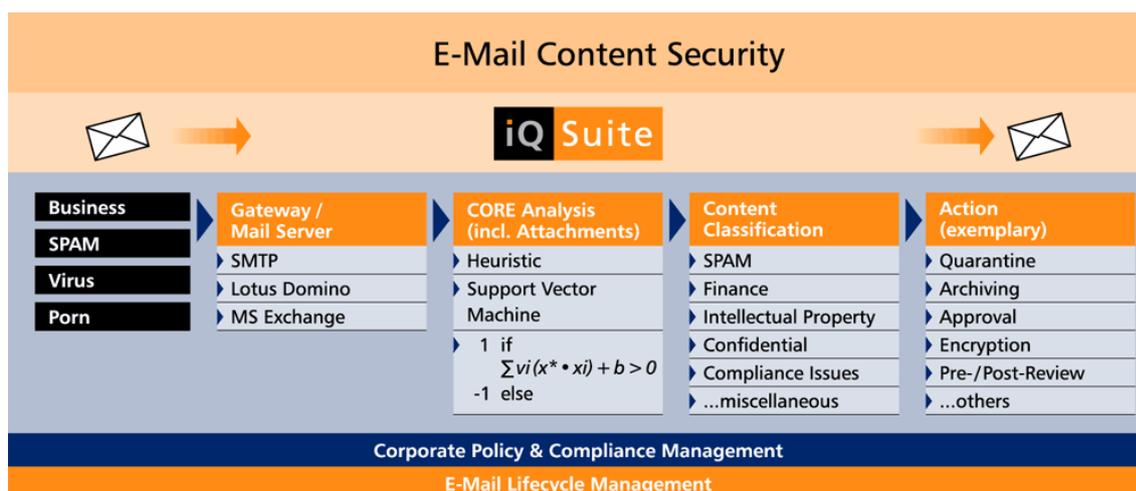


Illustration 2: Email Content Security

### 3 How Does the CORE Technology Work?

CORE is based upon the Support Vector Machines (SVM) method. SVM is a new generation of learning systems based upon progress made in statistical learning theory. SVM is a powerful method for practical applications, e.g. categorizing text or classifying images.

The goal of SVM in CORE is to optimize the classification of documents into specified categories. To achieve this goal, a classifier is trained using training documents. The documents used in training consist of a representative set of incoming and outgoing emails within a company (including spam). The more representative the selection, the better the method will work in actual operation.

Appropriate training documents come either directly from an email job that sends the mails to the training database, or else are copied and pasted to the training database from user mailboxes, safe archives, or quarantine databases. Forwarded emails are not suitable for training, as the forwarding information generated would also be incorporated into the training and could falsify the result.

#### 3.1 SVM Basics

The Support Vector Machines (SVM) method works with vectors. Each document is represented by a vector. To do so, a vector with the length “ $n$ ” is created from the quantity of the terms contained in the individual texts (quantity “ $m$ ”). The terms are all of the fragments contained in the document, e.g. words or HTML tags. Each individual document is then mapped to this vector, creating a vector space with the dimensions  $n \times m$ .

The individual terms are standardized and weighted using the TF-IDF method. “TF” stands for term frequency and represents the frequency of a word within a document. “IDF” is the inverse document frequency, and indicates the number of documents in which a term appears. The greater the frequency of a term within a text, the greater its relevance for classification. If the same term appears in many different documents, however, its importance for classification is reduced.

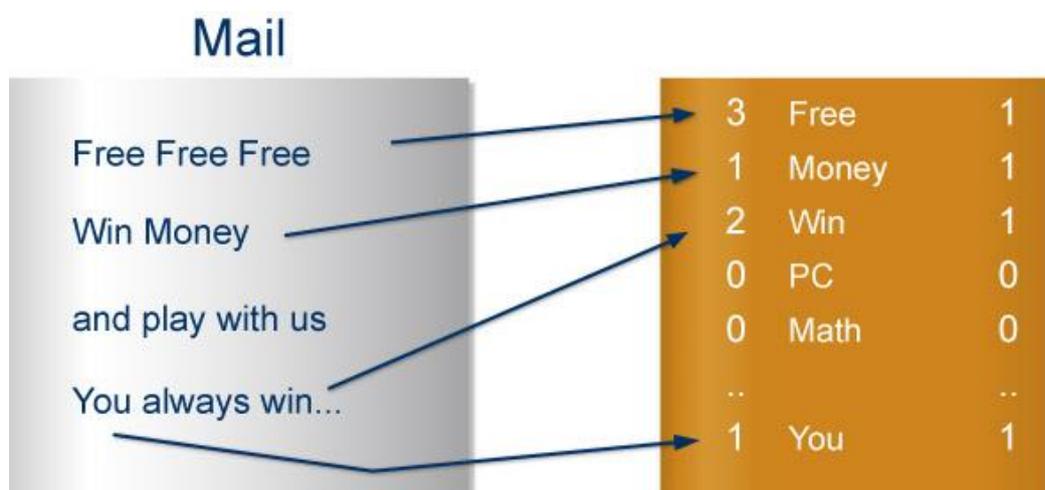
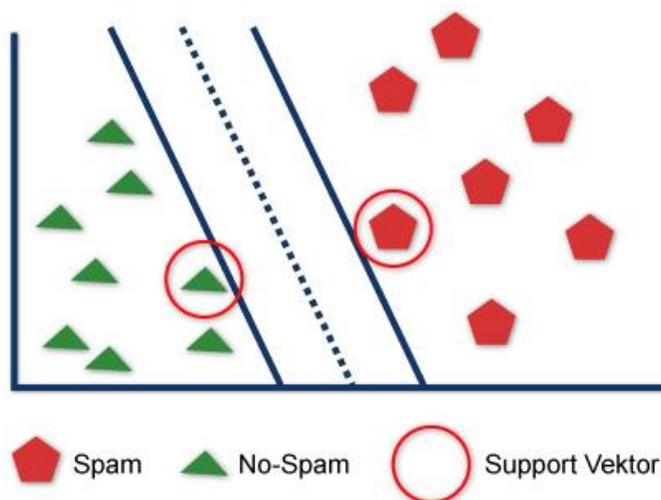


Illustration 3: Document vector example

Within the vector space, the SVM method calculates a hyper-level to create an optimal separation between the positive and negative training documents of a category. This also means that a linear optimization problem must be solved. The result is the class of training vectors that most closely approximate the hyper-level. These vectors are called support vectors.

In contrast to other vector classification techniques, the SVM method incorporates the complexity of the classifier in the algorithm. This prevents the trained classifier from becoming "overtrained" and thus able to correctly categorize the training documents only. This overtraining or "overfitting" can easily occur with naïve Bayes filters, which are easily susceptible because of their self-learning capability and lack of standardization.

The following figure shows the SVM mechanism in a two-dimensional space for the category "Spam."



*Illustration 4>: Structure of the hyper-level that maximizes the boundary between two classes*

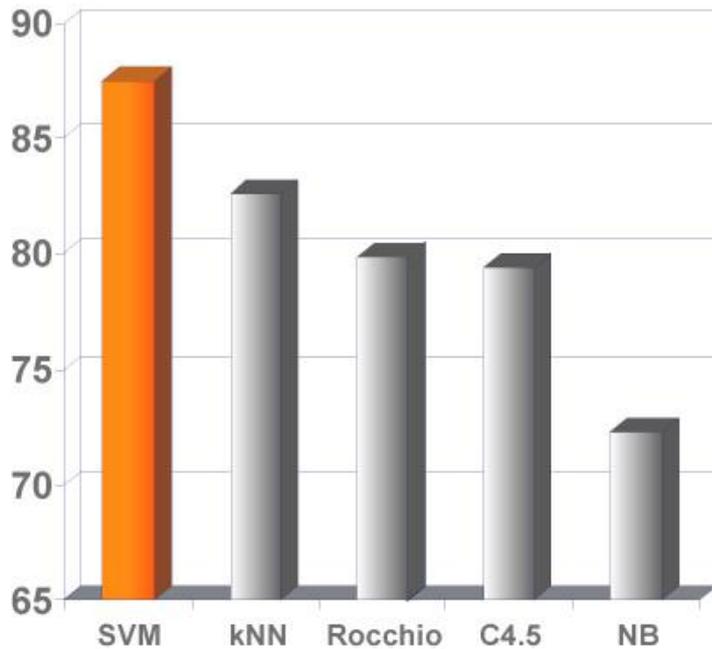
### 3.2 Why SVM?

The SVM method used in CORE offers the following advantages:

- SVM minimizes the number of classification errors
- SVM is robust against overfitting
- SVM performs very well, because efficient algorithms exist for solving the optimization problem
- SVM delivers very good classification results
- Text/content detection through the use of a modern statistical method
- Easy maintenance using the learning process
- Easy to adapt to company-specific email traffic using the learning process

SVM is the best currently available statistical method for text classification. Informatics specialist Thomas Joachims of Dortmund, for example, used the SVM method as far back as 1997 to classify the Reuters text collection with 86% accuracy across all categories and 91.4% accuracy across the 10 largest categories. The Reuters text collection is considered **the** standard benchmark for text classifiers. It consists of 9,603 documents in 118 categories.

### Number of correct classifications in the Reuters text collection



*Illustration 5: SVM "best in class" compared with other algorithms: kNN = k (number) Nearest Neighbors, Rocchio = Rocchio Algorithms, C4.5 = Ross Quinlan Algorithm, NB = Naive Bayes*

## 4 Practical Use of CORE for Anti-Spam

The following components are used for the effective deployment of CORE.

- iQ.Suite Wall as a basis for checking incoming emails and for the learning process.
- A reference set of emails that is categorized by the Administrator or other authorized persons.
- A training database as email container for the learning process.

### 4.1 Text Analysis with CORE

Initially, the text analysis with CORE is set up as follows:

1. Create a training database with a reference set of emails
2. Administrator or another appropriate person categorizes the emails
3. Configure the database training job
4. Run the database training job
5. Configure the database validation job
6. Run the database validation job
7. Configure the mail checking job - test operation
8. Run the mail checking job - test operation
9. Re-train or re-categorize the training database
10. Enable the mail checking job – live operation

## 4.2 Practical Tips

### 4.2.1 Categorization

- The Administrator or another authorized person first specifies the email categories to be defined. An email reference set, e.g. consisting of all emails for one business day, can serve as a basis.
- For an anti-spam application, only two categories are defined: SPAM and NOSPAM. For a more detailed distinction between different email types, it is also possible to define multiple categories, e.g. Newsletter, SPAM, Business, possibly also broken down by language.
- There must be at least 10 emails for each category. If you have defined multiple categories, it is better to restrict the training set for each category to approx. 25-50 typical mails than to use an excessively large quantity. With only two categories defined, each category should include at least 200 emails (better 500).
- For some newsletters and discussion forums, it may be reasonable to set up address exceptions in order to have these mails presorted, as CORE is likely to classify them as SPAM. If this is not possible, e.g. for Yahoo groups that always come from a different sender, then multiple categories (more than SPAM and NOSPAM) should be created from the reference set, with one category specifically for these groups.

### 4.2.2 Training and Validation

- The emails used for training must be sent directly from an email reference set into the training database, or moved there via copy and paste. In any case, no email header or forwarding information should appear in the message text, because this would falsify the text and thus the text analysis.
- A reference set of emails is generated most easily by creating a iQ.Suite Wall job that copies all incoming emails, e.g. during one day, into an additional quarantine database.
- HTML emails must be placed into the training database in their original form, as the HTML code is taken into account when the vectors are created.
- All emails for the training database should contain longer texts. To the extent possible, the "Business" category in particular should consist of emails with more than two sentences only.
- Training and validation jobs should be initiated manually and immediately checked for a successful run. They should never be run automatically at intervals.

### 4.2.3 Recategorization

- The *false positives* from the quarantine are sent to the training database via "Resend".
- If the result of recategorization is too many emails appearing in one category, then that category should either be divided, or the short emails should be removed from the training database.

The following illustration shows an example of categorizing an email reference set into multiple categories.

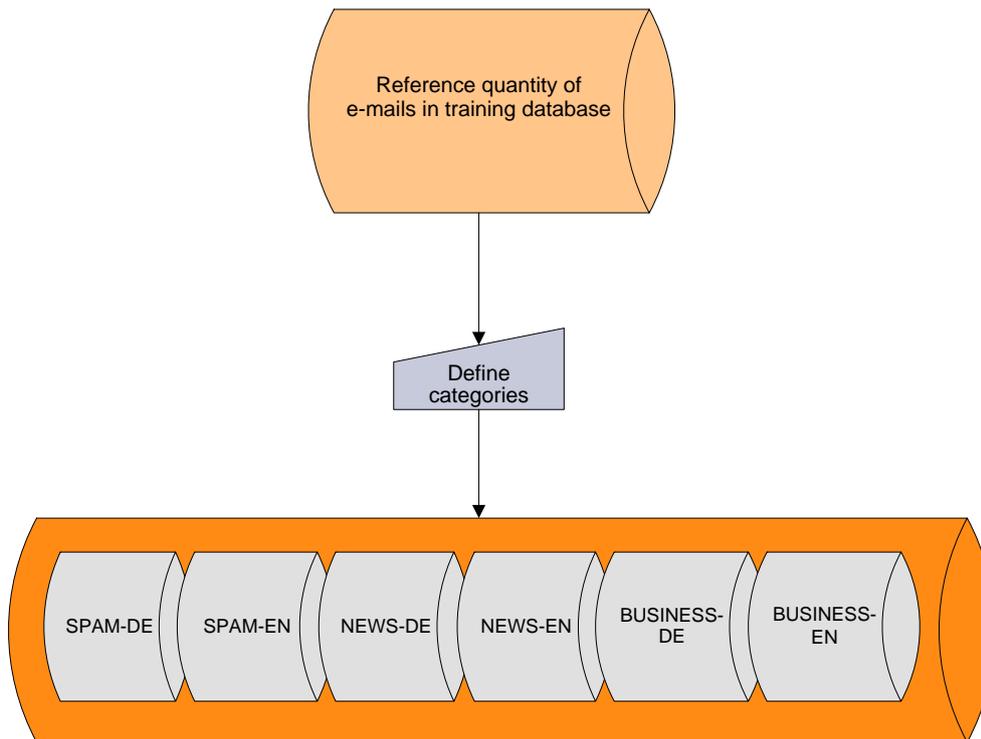


Illustration 6: Categorizing emails (example)

## 5 CORE for Anti-Spam – Highlights and Features

### Highlights

- Spam detection at its best

Adaptive processes are extremely well suited for the task of spam defense. Spam changes so rapidly that dictionary-based techniques and other lexical processes cannot keep up with user requirements.

- Integration into iQ.Suite

CORE is integrated seamlessly into the existing email checking sequence, without causing losses in performance.

- Latest technology for text classification

The Support Vector Machines (SVM) method used in CORE is the latest statistical method for text classification. It can be used for more than two categories without additional modules.

- Customized to company needs

Individual email categories with company-specific training documents enable optimal adaptation and independence.

In contrast to web-based spam services, email checking does not require a connection to external servers.

- Content analysis and document protection

Transparent management and monitoring of all incoming and outgoing email. Internal company documents can be classified and corresponding emails detected.

### Features

- CORE achieves more than 95% accuracy in classifying spam
- Fewer than 0.1% *false positives*
- Robust against overfitting
- Best statistical method for text classification in the Reuters text collection, with 86% accuracy across all 118 categories and 92% across the largest 10 categories
- Minimal number of classification errors (Structural Risk Minimization)
- Outstanding performance resulting from efficient algorithms that solve the optimization problem
- Multiple categorization possible
- Underlying SVM algorithm is freely available (Open Source) and is constantly being up-dated and optimized.
- Existing iQ.Suite Wall jobs serve as basis for checking
- New iQ.Suite Wall jobs serve as basis for teaching/learning process
- New Analyzer for checking documents
- Trainer integrated into the Analyzer for the teaching/learning process
- Document container for the teaching/learning process supports Copy & Paste

## **About GBS**

GROUP Business Software is a leading vendor of solutions and services in the fields of messaging security and workflow for the IBM and Microsoft collaboration platforms. Over 5,000 customers and more than 4 million users worldwide trust in GBS expertise. The company operates in Europe, North America and Asia.

Further information at [www.gbs.com](http://www.gbs.com)

**© 2016 GROUP Business Software Europa GmbH, All rights reserved.**

Our product descriptions are of a general and descriptive nature only. They do not stipulate any specific features nor do they represent any form of warranty or guarantee. We reserve the right to change the specifications and design of our products without notice at any time, in particular in order to keep abreast of technical developments. The information contained in this document presents the topics from the viewpoint of GBS at the time of publishing. Since GBS needs to be able to react to changing market requirements, this is not an obligation for GBS and GBS cannot guarantee that the information presented in it is accurate after the publication date. This document is intended for information purposes only. GBS does not extend warranty for this document, in either explicit or implied form. This also applies to quality, execution, standard commercial practice or suitability for a particular purpose. All the product and company names that appear in this document may be trademarks of their respective owners.