



V 1.2

Technical whitepaper Jun 2003

Rotterdam 24 Jun 2003

Contents

1. Introduction	3
2. Executive Summary	4
3. Product description	5
3.1. What it does.....	5
3.2. How it works: three components.....	5
3.2.1 Reception.....	6
3.2.2 Matcher.....	6
3.2.3 Executor.....	6
3.2.4 Resource Manager	7
3.2.5 Scheduler.....	7
4. Value adding modules	8
4.1. Planboard	8
4.2. Feedback.....	8
4.3. Web-enabled IVR-programming	8
4.4. Services module	9
4.5. SMS-module	9
4.6. Email-module.....	9
5. Interaction and integration	10
5.1. Interaction with existing systems	10
5.2. Integration in existing environment.....	10
Telephony integration	10
IP integration.....	11
6. Example applications.....	12
6.1. Heitje voor een karweitje (Bob-a-job).....	12
6.2. Falck Planning Application.....	12
7. More Information.....	13

1. Introduction

This document intends to describe the ASK system, its components and the requirements it imposes on the environment in which it is deployed. The intended audience is the IT and Telecom management.

The actual implementation process is not described, since it is covered in detail in the ASK installation manual.

2. Executive Summary

Modern Communication systems require modern technology. A larger part of our lives is dominated by the use of electronic devices for all sorts of tasks. Telephony is no exception. The integration of telephony and personal computing delivers a new level of functionality incorporating better relation management and improved accessibility control by automatically distinguishing caller and callee (requester and responder in the Ask terminology) and selecting an appropriate number to reach the required person.

The knowledge that can be gathered while using these communication systems is actively used by ASK to continually improve the performance of the organization. This quality is what sets ASK apart from the more mundane CTI solutions currently on the market.

3. Product description

3.1. *What it does*

ASK is a system that aims to provide a way to contact people by phone, email or other means based on an actual question. This question may relate to historical (I've been in contact with this person before), location (I need someone in the neighbourhood) or skill (I want to fix something and need expert advice) or a combination of this all and does not require actual address information. Instead a service number is provided that together with ASK determines with whom the person posing the question is connected. ASK provides a whole suite of mechanisms to implement all sorts of queries that can subsequently be used to build a particular service. It is e.g. possible to import calendar information or create it and use it to choose the appropriate phone number to use when contacting an individual. ASK is also equipped with the ability to check the perceived quality of its actions and use this information to improve on its behaviour.

Current supported communication methods are SMS, GSM, ISDN, Analog telephony, electronic mail and VOIP.

3.2. *How it works: three components*

The ASK system is based on a component framework. This facilitates future extensions and provides inherent scalability. ASK integrates a set of standard tools to provide an interface to the user account management and database maintenance, offering the opportunity to replace them by comparable programs if the environment so requires.

At the top level ASK consists of three components: Website, Database and the ASK Engine. These three components can be placed at one machine, but they can also be distributed across different machines. Each component can be duplicated and the components can still work together, because the architecture has been engineered to be able to work in a distributed environment. This makes it easy to realize a fully redundant setup, where the functionality of the system does not depend on machines or infrastructure. Experience taught us that machines have a certain risk of failure (for example a harddisk crash is often a weakness). Also infrastructure supplied by telephone companies and internet providers is not always as reliable as they want us to believe. With well-considered organization of the different redundant components of ASK the stability can be raised to desired levels.

The Website consists of a set of php-scripts, some Java-applets and supporting agents. The main task of this component is programming and configuration of ASK and enable online access.

The Database is a SQL-database and typically a MySQL-database, but can be, if desired, any other brand of database. We like to choose MySQL for a couple of reasons. The supported functionality of MySQL is reduced but satisfying. Advantage of this is the high performance MySQL can deliver. Besides that MySQL is build by the open source community which guarantees stability, less bugs and good support (although this may sound strange). That this software is free of license costs becomes therefore only a side issue.

The third component, the ASK Engine, consists of a set of software agents that do the real job. The agents are in effect programs in their own right, that communicate using a proprietary protocol designed to be extended, this also ensures that the agents are not bound to one machine but can easily distributed across a network. To allow maximum scalability the agents each spawn some threads to achieve more simultaneous functionality if necessary. 5 major agents are discussed here to give some view on how ask works. The first 4 described agents work in a pipeline to get high performance.

3.2.1 Reception

This is where the incoming request is processed and classified. Based on the identification information, like the telephone number somebody is calling with or the email address used, the request can be forwarded to the right ...

3.2.2 Matcher

The Matcher receives the service request from the Reception. Matching can be quite difficult, because both the demands of the requester and the responder have to be met. The Matcher searches for a 'Top-five List of responders' who can reply best to the request. This can be based on different kinds of matching algorithms. Matching algorithms are for example:

- **Round Robin**; randomly choose a responder that is available.
- **Last Spoken**
- **Rating**; by using feedback some rating info is gathered. The requester is connected to the best responder.
- **Friendly Rating**; the same as rating but the requester sometimes gets transferred to a responder who does not have the highest rating. This could be necessary for some applications to give responders the ability to increase their rating.

In general the matching algorithm can be slightly adjusted per application because the relevancy of the classification criteria will be different in every single case.

After determining the best possible candidate the matcher informs the ...

3.2.3 Executor

The executor is responsible for actually connecting the requesting individual with the identified responder. The responder profile contains the location if fixed, the classification of available knowledge and skills, the availability for different levels of emergency, the history of calls including user feedback and if relevant the pricing information.

When a possible connection has been identified, it has to be established. The Matcher therefore sends the completed request back to the ...

3.2.4 Resource Manager

The Resource Manager is the agent that is responsible for the actual connection with the communication medium. So if the incoming request is done by a GSM and the responder has a GSM too which is available then the resource manager builds the connection between the 2 GSM's. By using a separate agent for this task, ASK ensures that all the other agents are independent from the actual communication used. The resource manager now supports; analogue telephony, ISDN telephony, GSM, Voip, SMS and Email.

3.2.5 Scheduler

The scheduler is a stand-alone program. It is almost completely responsible for the self learning/organising capabilities of ASK. It usually takes some action after a connection has been completed. So if a requester has been connected to a responder and they both closed the connection the scheduler may connect to one (or both) of them and get some feedback. The scheduler is also capable of ensuring that a minimum availability rate is attained. If for example nobody is available as a responder it's possible to let ASK automatically call some responders and ask them to get available.

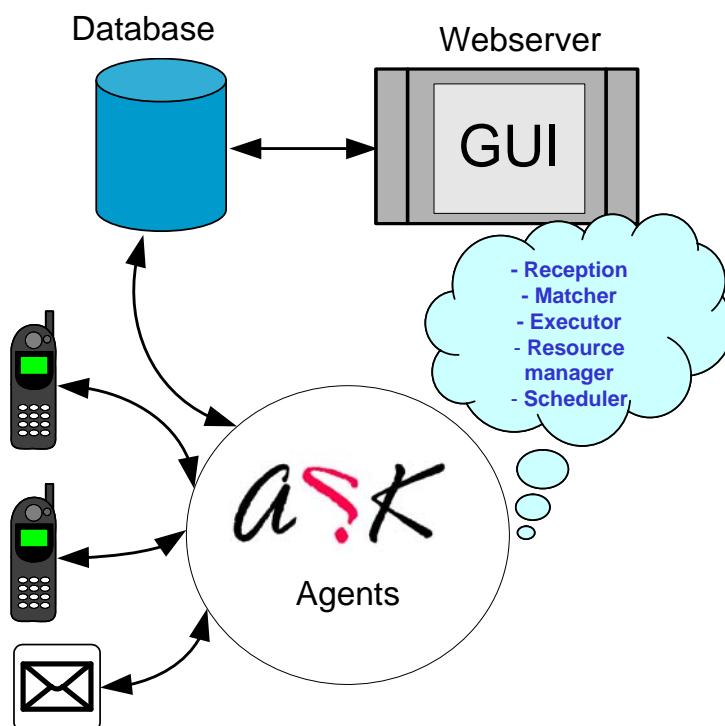


fig1. the ASK architecture

4. Value adding modules

ASK is being shipped with modules that increase the utility and flexibility of the product. They make ASK more useful for a wide range of applications, whether this concerns Care, Planning or Customer Service. Which modules will be employed, depends on the environment in which ASK will be used.

4.1. *Planboard*

The planboard allows users to organize their availability and reachability. Because the plan board is built on standard technologies like php and Java, it is possible to access the plan board from any place in the world using the Internet.

4.2. *Feedback*

Feedback is a mechanism in ASK to realize the self-organizing aspect. This is in the ASK philosophy a very important aspect. Depending on a set of programmed constraints, ASK will automatically acquire feedback by calling the requester/responder or by sending him an email or SMS. The rating that one user has given to another will be stored in the database. If ASK makes decisions about connecting requesters afterwards, all the ratings retrieved in history are used to realize a 'Best Match'. Through this relatively simple mechanism ASK learns what requesters want and what they don't want. And users will notice that they are almost always connected to the right person at once.

4.3. *Web-enabled IVR-programming*

Another module of ASK is the IVR-module. We are used to call this module Betsy, an automated receptionist, because this module can take over tasks that are typically handled by the receptionist of a company. She asks requesters what they want and tries to satisfy their needs as best she can. The module has the possibility to leave a message for possible responders or to put them through to a person, department or service. How Betsy will respond to certain requests, does not only depend on what phone number is called, but also on who is calling, which group the caller is from and on what the current availability is of the person that is requested for. Betsy can be programmed by using a drag and drop web-interface allowing maximal flexibility.

4.4. Services module

The Services Module is used to create and manage services. A service is in fact a number of requesters/responders organized in a group, that share some common properties. These groups contain requesters/responders that are responsible for running the service. One mechanism that can be used within a service is the escalation mechanism. By default the service is employed by a basic group of users, but this mechanism will escalate to the first escalation group when the users of this basic group fail to respond. Even a second escalation group can be defined where generally a management group is put so they can respond for the service and in the meantime get informed that the first two groups failed to respond.

A second mechanism within the Services Module is availability tracking. A minimum level of availability can be set. At the moment that the mean availability of all users that are responsible for the service drops under the minimum, ASK will take action by calling users and ask them to become available. By using this mechanism it can be ensured that a service always meets the requirements.

4.5. SMS-module

The SMS-module allows responders who are (temporarily) not able to change their availability by using the Internet, to use SMS for this instead. Other types of actions can be done by using SMS as well. In this manner ASK and the people registered at ASK can still be reached in an offline environment by simply using their phone.

4.6. Email-module

ASK also uses email to communicate to users. It is possible to address an email message to a service that will then try to deliver it where it can be handled. Email is also extremely flexible as an add-on to supply error handling and generally off-line functionality.

5. Interaction and integration

5.1. Interaction with existing systems

It's often the case that a company already uses some kind of an information system for example to keep track of availability of objects or telephone numbers of employees. Because of the database structure used by ASK this isn't a problem. ASK stores and reads all important information in its own MySQL database. It can easily read the contact information or the reachability data from an other foreign database. In a number of scenarios ASK has to take action when a certain problem arises, there is a special agent for this. This agent, the scheduler, continuously reads the foreign database and if ASK should take an action it will do that and can even log the taken action in the foreign database.

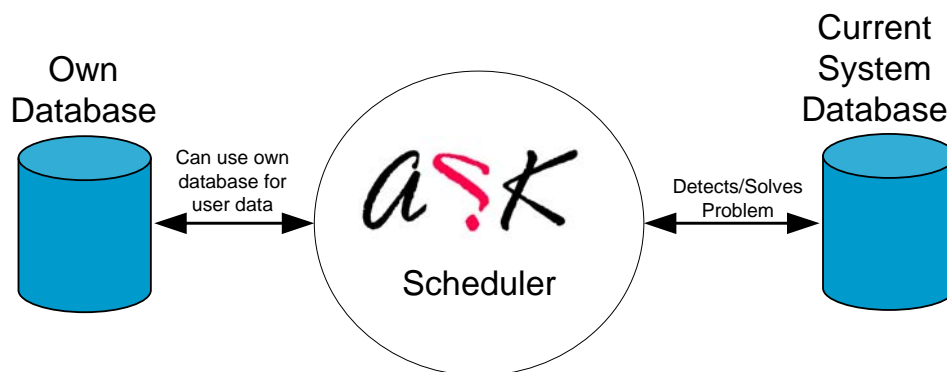


fig 2. Interaction with another system

5.2. Integration in existing environment

Most companies are already using some telecommunication infrastructure like a central exchange. ASK is designed to cooperate with such systems in a number of ways. It can even coexist with most implementations to guarantee an easy and safe switch. This is of course only necessary if there is an existing telecommunication infrastructure, because ASK can do the telephone traffic routing itself.

Telephony integration

The most used configuration is shown below. Only some specific ASK phone numbers are routed directly into ASK. And all the other telephone traffic is handled on the old way. Ask supports most central exchanges but there are some requirements; the central exchange should have an ISDN interface and all the internal telephones should be reachable for the central exchange.

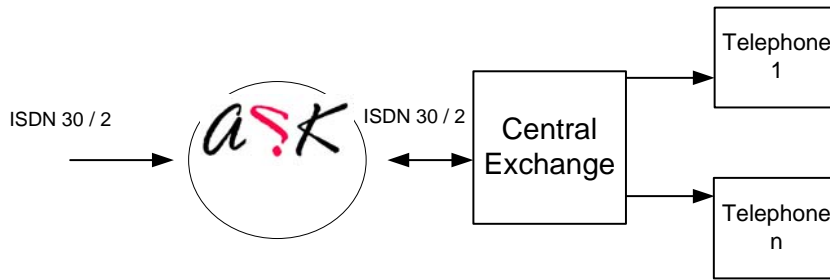


fig 3. Ask cooperating with a Central Exchange

IP integration

To make ASK easy to administrate all the configuration can be done using the web interface. Because the ASK box is located inside the company's LAN there is no need for an internet connection.

6. Example applications

In contrast with all other chapters which tell you the story about ASK, but kept at a rather abstract level, we here explain two of our projects we deployed in the early days of ASK. The projects described here are meant to give you a basic view of how ASK can be deployed.

6.1. *Heitje voor een karweitje (Bob-a-job)*

In today's big city's more and more senior citizens get socially isolated. These seniors experience trouble contacting other people because of their differences in language, age and culture. To keep these people active and less socially isolated an old Dutch tradition has been brought back to life; Bob a job, which means a small job for little money. (All by telephone). Seniors from the West District in Rotterdam can already call one central number, where all chores are divided among young volunteers. Once ASK is installed these calls can be send directly to the volunteers' mobile phones, classified on history, skills or location.

6.2. *Security Business Planning Application*

It is the security business their mission to provide assistance to people, authorities and companies in order to meet their demands for safety and security as individuals or groups on an day to day basis. It's the everyday basis that consumes lots of time for a good planning. Mostly the planning is done some weeks before the actual deployment. It costs a lot of time when somebody gets ill or wants to switch a shift to get the planning current. They use a specially designed application to make the planning. Now, with ASK, the employees can call (or sms) a central telephone number and then ASK tries to connect him with other employees that could fill in the shift. If the other employee accepts the shift then ASK will fill in the shift and updates the planning application they already used. Another scenario is now also possible. A planner makes a gap (manually) in the planning database (he deletes the shift of somebody who should be working tomorrow). ASK detects the gap in the planning system and phones some other employees to see if they can fill in the shift.

7. More Information

For more information of ASK you can visit the corporate website www.ask-cs.com. For technical questions about implementation or cooperation with current systems you can contact our technical staff by sending an email to rene@ask-cs.com.