Chapter 2

Situation and Requirements

Requirements

If an R/3 System is used in different countries, then a company may need to manage several languages and codepages. Fig. 2-1 depicts a sample configuration in an R/3 System. The three languages Japanese (with codepage S-JIS), German (with codepage Latin-1), and English (contained in both codepages) are supported in the system. Japanese users who are logged on at instance 1 wish to input Japanese addresses. German users logged on at instance 2 also wish to maintain addresses. All data is stored in one database (in the middle of Fig. 2-1). Another task is to print documents which contain characters from different languages (with different codepages; see at the top of Fig. 2-1).

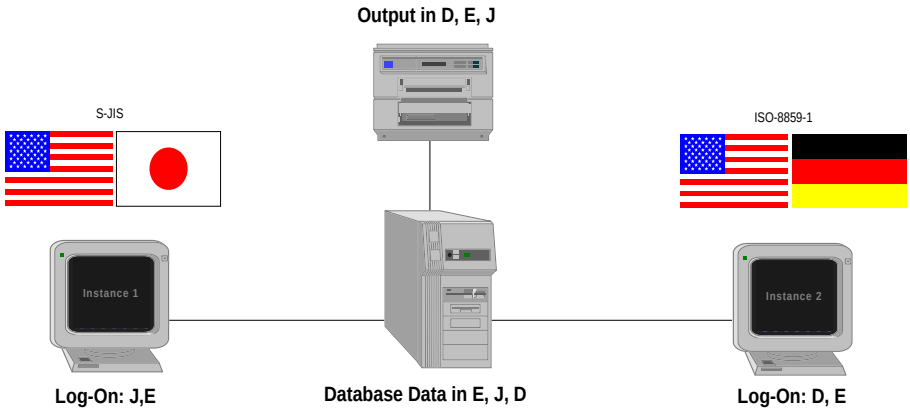

Fig. 2-1: Multi Language Support in an R/3 System

Single-Codepage System

The R/3 System is currently implemented as a multi-language single-codepage system. An R/3 System can manage all languages (i.e. the characters of the languages) that are defined in one codepage. Today's implementation must be extended, if the R/3 System is to be able to manage two or more languages from different codepages (e.g. German and Japanese).

Database

In the R/3 System, the database serve uses a single-byte codepage and the table entries are sorted bitwise. SAP does not support the use of more than one codepage in the database server.

Double-Byte Codepages

In addition to the single-byte version, there is also an implementation of the R/3 System for double-byte codepages. For example, the codepages EUC and S-JIS are used in the Japanese double-byte implementation. The double-byte implementation always supports an Asian language and English (e.g. Japanese and English, Mandarin and English). In general, it is not possible to mix Asian languages in one double-byte codepage implementation (e.g. Japanese and Chinese). A mixture of single-byte and double-byte codepages also is not supported (e.g. Japanese and Latin-1).

Mixture of Codepages

In current R/3 installations which need to solve the multi-codepage problem, different codepages are used in the system components (SAP frontend, application server, database). Fig. 2-2 shows a distributed R/3 System with a central database server at the bottom, some application servers, and the connected SAP frontends at the top. Each of these components can use a different codepage. The codepage of the database server in the R/3 System is irrelevant because the table entries are sorted bitwise. There is no problem if an application server accesses data in the database using the same codepage as the application server that wrote the data into the database. If the codepages differ, data can be lost because the characters are interpreted incorrectly. In addition, it is not possible to translate the data from one codepage to another, because the data is stored in the database without codepage information.

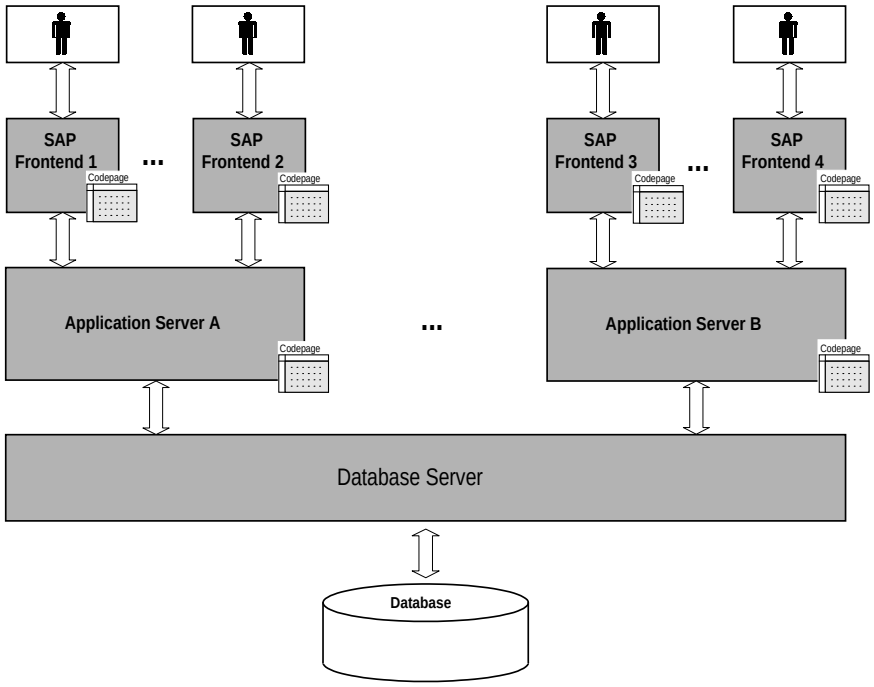

Fig. 2-2: Codepages in an R/3 System

Translation of Codepages

An SAP frontend can use a different codepage than the application server to which it is connected. In this case, a translation of the data between the two codepages must be done when data is sent from the application server to the SAP frontend or vice versa. But such a translation does not always make sense (e.g. translation from Latin-1 (German) to EUC (Japanese)). Furthermore, data can be lost during a translation (e.g. EUC to special OS/2 codepage). If an application server sends a character to the frontend and no unique assignment for the character exists in the frontend's codepage, then the character can be falsely interpreted in the frontend, with the consequent data loss when the character is returned from the frontend to the application server.

Solutions

The aim of this report is to present a solution that fulfills the requirements on an R/3 System with multi-codepage capability and that eliminates the problems described above. The next chapter illustrates two solutions: The first describes an implementation with Unicode. However this solution is not feasible at this time. The second solution shows the implementation planned for release 3.0 (and 2.2 F).

Existing Solution with Distributed Systems

Consolidation

An existing solution for companies that are active worldwide is distributed systems (see Fig. 2-3) in which one system exists for each language (or at least for each codepage). Each system contains data that must be consolidated globally. For that purpose, central consolidation systems (right side of Fig. 2-3) are installed. To be able to manage global data in a consolidation system, the data must be available in every system in a character set that is common to all systems. Therefore, a set of characters (*common character set*) is defined which is the value set for global data.

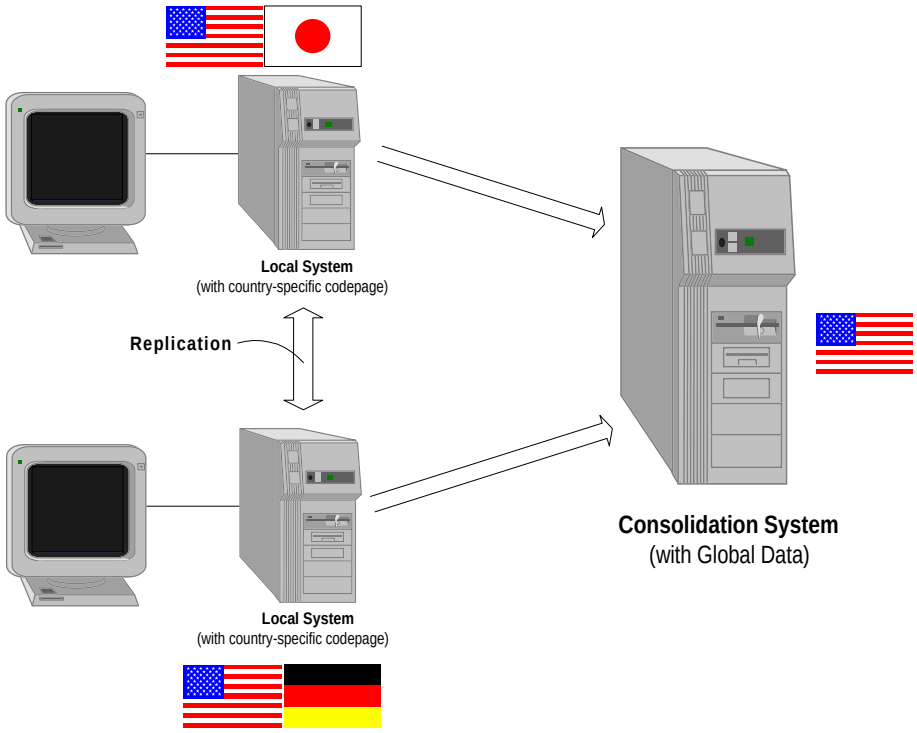

Fig. 2-3: Consolidation in Distributed Systems with Different Codepages

Replication

Global data which are created in a source system must be replicated in other systems (see connection between local systems in Fig. 2-3). For this replication special transport mechanisms must exist to perform the transfer from one system to another.

next chapter